DE 88015370

UCRL-53875

Allocating and Launching MaRV Interceptors Based on Miss Distance Predictions

G. C. Corynen

June 1988

Lawrence Livermore National Laboratory

DISCLAIMER

This document was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor the University of California nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial products, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or the University of California, and shall not be used for advertising or product endorsement purposes.

Work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract W-7405-Eng-48.

UCRL-53875
Distribution Category UC-32

Allocating and Launching MaRV Interceptors Based on Miss Distance Predictions

G. C. Corynen

Manuscript date: June 1988

LAWRENCE LIVERMORE NATIONAL LABORATORY
University of California • Livermore, California • 94550

Available from: National Technical Information Service • U.S. Department of Commerce
5285 Port Royal Road • Springfield, VA 22161 • A04 • (Microfiche A01)

CONTENTS

List of Figures

Allocating and Launching MaRV Interceptors

Based on Miss Distance Predictions

G. C. Corynen

Abstract and Summary

We have developed battle management algorithms for assigning and launching ground-based interceptors to neutralize a threat consisting of multiple decoyed MaRVs and BRVs in their terminal phase.

Decisions are based on the minimization of expected leakage risk, and they specifically account for uncertainties and errors in the identity of each threat object, although detailed discrimination issues are not discussed. In contrast to standard probabilistic approaches, the algorithms are physics-based and use kinematic and geometric models for estimating interceptor-MaRV miss distances and probabilities of kill to determine optimal waiting times before launch.

Based on a test comparing interceptor and MaRV technologies, the minimax algorithm finds the waiting times for which the minimum probability of kill is a maximum. To avoid a full multidimensional optimization, we use a "greedy" one-dimensional approach where only the most urgent threat is addressed optimally, conditional to allocating to the remaining threats (dimensions) a nominal quantity of interceptors determined by previous optimizations. One-on-one engagements are thus emphasized, and the preponderance of the work discussed in this report was directed towards the development of a miss distance prediction algorithm, called the MD Algorithm.

While functional and software testing of the whole battle manager is still under way, experiments with the MD algorithm have been completed and have provided useful insights which generally agree with conclusions reached by others. It was found that miss distances are not very sensitive to relative interceptor-MaRV velocities but are quite sensitive to relative maneuvering capabilities, as measured by relative lateral acceleration limits. More specifically, for MaRVs and interceptors with conventional capabilities, and for a total defense system response time of 0.1 second, the minimax miss distance was predicted to be about 600 feet.

Chapter 1

INTRODUCTION

Recent technological advances in the design of Ballistic Re-Entry Vehicles (BRVs) have stimulated an increasing interest in the effectiveness of *Maneuverable* Re-Entry Vehicles (MaRVs), and in the development and utilization of vehicles designed to intercept MaRVs. The research documented in this report is part of a comprehensive effort to develop modeling and simulation software to analyze terminal engagements between interceptor defense sites and multiple decoyed MaRVs and BRVs. The development of this capability requires the solution to various interceptor resource allocation and launch logic problems, and in this report we address two such battle management problems:

1. The assignment of interceptors to MaRVs and BRVs,
2. The selection of interceptor launch time and direction.

Significant contributions have been made by others in the development of strategies to manage terminal engagements with MaRVs (see References [1–7], for instance). But these and other current approaches are mostly probabilistic and do not sufficiently reflect the physics of engagements, hence they are often too abstract. In contrast, the approach presented here uses kinematic and geometric engagement models to predict interceptor miss distances for different launch delay times. By varying these delay times, miss distances can be minimized and kill probabilities maximized. This results in optimal interceptor resource allocations, and in the minimization of defense leakage risk.

Although we impose no specific restrictions on the quantity of interceptors and attacking objects, we have emphasized one-on-one engagements, for two major reasons. First, collective interactions between objects turn out to be rather weak compared with individual interactions between one interceptor and one attacking object, particularly when that object is a MaRV. Second, these one-on-one interactions also turn out to be relatively more complex and computationally challenging than collective ones, at least in our current battle management framework.

Similar to judging the performance of a defensive system by the proportion of threats which leak through it, the performance of an interceptor, and its associated launch policy, may be judged by the probability that the interceptor will neutralize the threat to which it was assigned. Because engagements typically take place a considerable time after launch, the battle manager must predict this probability, sometimes far in advance, to determine *when* to launch, and to decide *how many* interceptors should be assigned to the threat. Considering that the kill probability of an interceptor is very sensitive to the miss distance achieved by it, a capability to predict miss distances is essential to an effective battle manager. This prediction problem is the principal subject addressed in this report.

Our discussion is structured as follows. In the remainder of this chapter, we summarize the battle managment problem and present our general solution to it. In particular, we solve the first problem listed earlier: that of assigning interceptors to MaRVs and BRVs. Subsequent chapters are devoted to solving the second problem of selecting the time and direction of interceptor launch. In Chapter 2, we present the mathematical preliminaries required to discuss this second problem. In Chapter 3, we present our miss distance prediction and minimization methodology, and in Chapter 4 we report the result of computational experiments with the miss distance algorithm. Conclusions and future work recommendations are made in Chapter 5. Nine comprehensive appendices round out the report.

1.1. The Battle Management Problem

A simplified pictorial of the general battle management problem considered in this paper is described in Figure 1.1, where a set of *threat objects* $\mathcal{O} = \{O_q\colon\ q = 1, \ldots, Q\}$ is threatening a collection of assets $\mathcal{A} = \{a_k\colon\ k = 1, \ldots, K\}$ which are protected by pools of interceptors $\mathcal{I} = \{I_u\colon\ u = 1, \ldots, U\}$ located at different sites. Threat objects are of two basic types: MaRVs and BRVs. Interceptors are assets which can be members of several pools. They are also of two basic types: Overlay Interceptors (type 1), and Underlay Interceptors (type 2).

As indicated earlier, the major objective of the battle manager is to make the best use of available resources to defend its assigned assets. This is a problem in optimal decision making which involves three principal subproblems, as shown in Figure 1.2. In practice, the battle manager rarely knows with certainty that the incoming objects or threats are in fact MaRVs and, in the next sections, we discuss how these types of uncertainty are incorporated into our framework. But we do not discuss detailed discrimination issues.

Referring to Figure 1.2, threats must first be prioritized so that "important" threats can be processed first. This prioritization is particularly important in threat-rich situations where only a subset of the threats can be processed by the battle management computers during one update cycle, and we discuss it in greater detail in Section 1.3.

Next, the battle manager must decide how many interceptors should be allocated to each incoming threat, and which type of interceptor should be used. In some schemes, this is accomplished simply by specifying a desired kill probability, and allocating a sufficient number of interceptors to achieve this specification. In our scheme, we also allow this

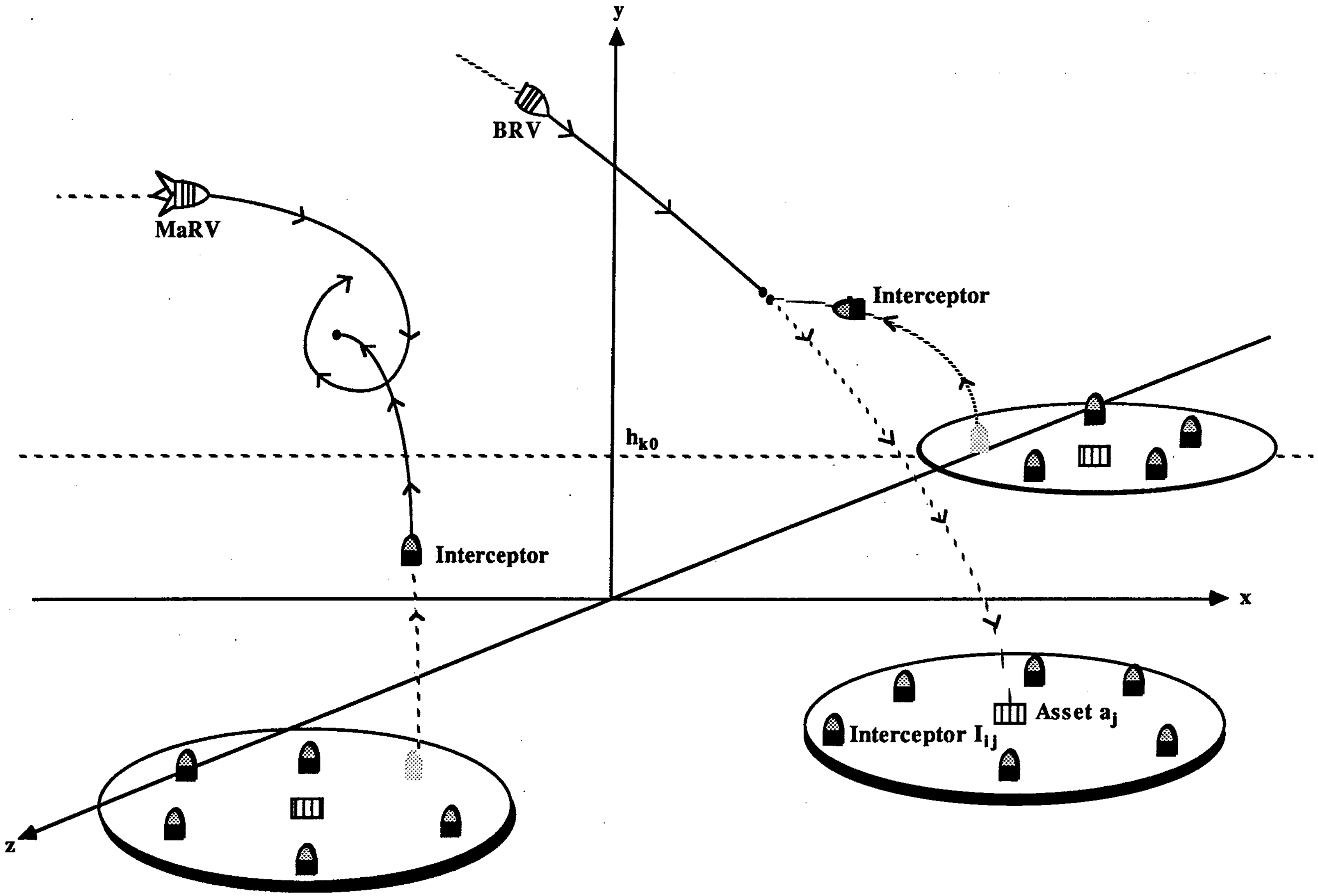


Figure 1.1. Typical engagement geometry.

approach, but generally take the broader view that such decisions involve several additional parameters relating to defense attrition, regret, and so on. This is discussed further in Section 1.2 below.

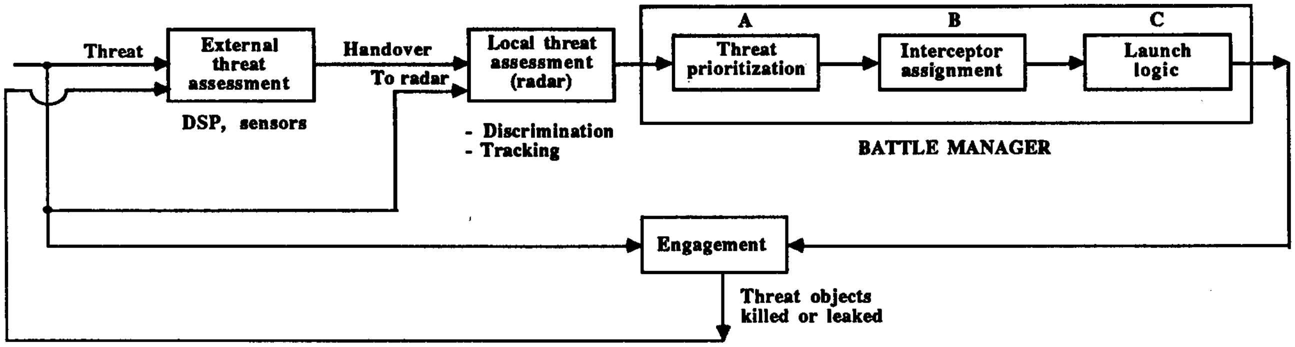


Figure 1.2. The battle manager performs three major functions.

In the interceptor-launch logic subsystem, the battle manager decides the direction and time of interceptor launch. This is the problem emphasized in this report, and we discuss it in Chapters 2 through 4.

1.2. Battle Management Decision Policy

In assigning interceptors to threat objects, the battle manager must achieve an appropriate balance between the costs resulting from threat leakage and the expenditures required to prevent such leakage. Many of the parameters determining these costs and expenditures are imperfectly known. Leakage costs, for instance, depend on the type and size of the threat and on the vulnerability and lethality of threat objects and defended assets. Defense costs are dominated by the intrinsic ("lifecycle") costs of interceptors and by the risk of reduced protection associated with the launch of interceptors from a limited inventory.

We approach the battle management problem as a stochastic optimization problem where an optimal balance is sought between leakage risk and defense cost. For this optimization, we use *total expected loss L* as the objective function. More specifically, our goal is to minimize the loss function

$$L_T(m_{\alpha q}) = \sum_{q=1}^{Q} \left(L_q \cdot p_{L_q}(m_{\alpha q}) + C_q(m_{\alpha q}) \right) \quad , \tag{1.1}$$

where L_q is the loss incurred if object O_q leaks through,

p_{L_q} is the probability that O_q leaks through,

C_q are the costs incurred by allocating defense resources to O_q,

$m_{\alpha q}$ is the quantity of defense resources of type α allocated to O_q.

This is a Q-dimensional constrained optimization problem where an optimal vector $m^*_\alpha = [m^*_{\alpha 1}, \ldots, m^*_{\alpha q}, \ldots, m^*_{\alpha Q}]$ must be found, subject to the constraint that defense resources of type α cannot exceed U_α:

$$\sum_{q=1}^{Q} m_{\alpha q} \leq U_\alpha \quad . \tag{1.2}$$

Although this chapter is only a summary of the battle management approach, a few comments should be made to demonstrate the generality and flexibility of the objective function in Equation (1.1).

Even though the additive structure of this function might suggest that the probabilistic events and random variables involved should be either disjoint (mutually exclusive) or independent, this is not the case, as shown in Reference [8], p. 12. Additivity or linear "accumulation" of the losses L_q are also not required. All we need is to assign or to estimate a loss L_q for each threat object O_q. Admittedly, if several O_qs are headed for the same asset (target), these loss values may depend upon which O_q reaches its destination first, but that is an assessment problem, not a problem in structuring the loss function of Equation (1.1). Stepping away from the mathematics for a moment, the battle manager assesses these L_q values in real time, based on radar and other sensor updates concerning the position, direction, and destination of the associated O_qs. If several objects are perceived by the sensors to be headed for the same target, this information will be used to estimate which object will reach the target first, the damage caused by that object, and any additional damage which will be caused by the remaining objects still in flight.

Whereas $m_{\alpha q}$ can be used to indicate many different types of resources such as warhead yield, booster capability, payload, and so on, we shall use it exclusively to specify the quantity of interceptors of type α allocated to threat O_q.

Turning our attention to the *computation* of Equation (1.1), we shall see that p_{L_q} and C_q are nonlinear functions of $m_{\alpha q}$, and these parameters also share common variables with L_q. For instance, all three parameters depend on whether threat object O_q is a MaRV or a BRV. Hence the optimization problem cannot easily be twisted into a linear programming problem, unfortunately. Considering the computational implications of attempting to solve the full nonlinear problem, we have chosen a more myopic or "greedy" approach. Rather than solving the full Q-dimensional problem, we consider one threat object O_q at a time, starting with the highest priority one, and we attempt to find the optimal value $m^*_{\alpha q}$ for that threat, invoking some mild assumptions about the future treatment of the remaining $Q-1$ threat objects. More specifically, we shall assume that BRVs down the priority list are assigned a single interceptor of the appropriate type, and that MaRV's are assigned two, one of each type, until the limit U_α is reached. This assumption sets a "level" for the local optimization where the optimal value $m^*_{\alpha q}$ for the highest-priority O_q is sought. This one-dimensional approach obviously encourages iteration over different levels. If we find, for instance, that $m^*_{\alpha q}$ consistently exceeds 2, the nominal allocation level of the remaining objects should probably be increased, but this can easily be determined experimentally on the computer. In the worst case, this "local" scheme may not approach the true optimum—as obtained by Q-dimensional optimization—and a more sophisticated approach may be necessary. This will be established with exhaustive tests when the battle manager is fully implemented.

In the next two sections, we discuss each of two components of the objective function in greater detail.

1.2.1. COMPUTING LEAKAGE LOSSES

We estimate the costs of threat leakage by the *expected* asset loss resulting from the leakage. If L_q is the loss incurred, conditional to object O_q leaking through, and p_{L_q} is the probability that O_q leaks through, the expected leakage loss is

$$L = \sum_{q=1}^{Q} L_q \cdot p_{L_q}(m_{\alpha q}) \quad , \tag{1.3}$$

where we emphasize the fact that the p_{L_q} depends on the quantity $m_{\alpha q}$ of interceptors of type α assigned to threat O_q by the battle manager. While L may include *interceptor* losses due to leakage, it does not account for the loss of protection arising from the assignment and launch of interceptors. Such losses will be discussed below, and in Section 1.2.2.

But L_q and p_{L_q} both depend on the type of the threat object O_q. This introduces dependencies between p_{L_q} and L_q, and requires the modification

$$\begin{aligned} L &= \sum_{q=1}^{Q} \left[\left(L_q p_{L_q} |_{O_q=M} \right) p_q^M + \left(L_q p_{L_q} |_{O_q=B} \right) p_q^B \right] \\ &= \sum_{q=1}^{Q} \left(L_q^M p_{L_q}^M p_q^M + L_q^B p_{L_q}^B p_q^B \right) \quad , \end{aligned} \tag{1.4}$$

where $p_q^M \; +p_q^B + p_q^D = 1$,

p_q^M is the probability that O_q is a MaRV,

p_q^B is probability that O_q is a BRV,

p_q^D is the probability that O_q is a decoy,

$L_q^M(L_q^B)$ is the leakage loss if O_q leaks through and is a MaRV (BRV),

$p_{Lq}^M(P_{Lq}^B)$ is the leakage probability if O_q is a MaRV (BRV).

But L_q^M and L_q^B depend on the probable destination and reach of O_q, and on the assets threatened. Hence

$$L_q^M = \sum_{k=1}^{K} V(a_k) p_{qk}^M \tag{1.5}$$

and

$$L_q^B = \sum_{k=1}^{K} V(a_k) p_{qk}^B \quad , \tag{1.6}$$

where $V(a_k)$ is the value of asset a_k,

p_{qk}^M is the probability that O_q is headed for a_k if O_q is a MaRV,

and similarly if O_q is a BRV.

In our framework, we estimate or approximate p_{qk} with the probability that asset a_k lies in the footprint f_q of O_q (see Reference [9] for a discussion of footprint calculations). Thus,

$$p_{qk}^M = p_q\left(a_k \in f_q|_{O_q=M}\right) \tag{1.7}$$

and

$$p_{qk}^B = p_q\left(a_k \in f_q|_{O_q=B}\right) \quad . \tag{1.8}$$

where f_q is the footprint of O_q (a random function).

Next, considering $m_{\alpha q}$ interceptors assigned to a single O_q, and assuming that the interceptors damage O_q independently,

$$\left(p_{L_q}(m_{\alpha q})|_{O_q=B}\right) = \left(p_{L_q}(m_{\alpha q}=1)|_{O_q=B}\right)^{m_{\alpha q}} \tag{1.9}$$

and

$$\left(p_{L_q}(m_{\alpha q})\Big|_{O_q=M}\right) = \left(p_{L_q}(m_{\alpha q}=1)\Big|_{O_q=M}\right)^{m_{\alpha q}} \quad . \tag{1.10}$$

Using the simple facts that

$$p_{L_q}(m_{\alpha q} = 1)\Big|_{O_q=B} = \left(1 - p_{K_q}(m_{\alpha q} = 1)\Big|_{O_q=B}\right) \tag{1.11}$$

and

$$p_{L_q}(m_{\alpha q} = 1)\Big|_{O_q=M} = \left(1 - p_{K_q}(m_{\alpha q} = 1)\Big|_{O_q=M}\right) \quad , \tag{1.12}$$

where $p_{K_q}(m_{\alpha q} = 1)$ is the probability that an interceptor of type α will kill O_q, we can combine Equations (1.1)–(1.12) to obtain the overall leakage loss

$$L = \sum_{q=1}^{Q} \sum_{k=1}^{K} \left[V(a_k)\left(p_{qk}^M p_{L_q}^M(m_{\alpha q}) p_q^M + p_{qk}^B p_{L_q}^B(m_{\alpha q}) p_q^B\right)\right] \quad , \tag{1.13}$$

where

$$p_{L_q}^M(m_{\alpha q}) = \left(1 - p_{K_q}^M(m_{\alpha q} = 1)\right)^{m_{\alpha q}} \quad , \tag{1.14}$$

and Q is the quantity of threat objects,

K is the quantity of assets,

$m_{\alpha q}$ is the quantity of interceptors of type α launched simultaneously towards O_q,

$p_{K_q}^M(m_{\alpha q} = 1)$ is the single-shot kill probability for a MaRV threat O_q,

and similarly for the BRV superscript B.

The individual kill probabilities $p_{K_q}^M$ and $p_{K_q}^B$ used in Equations (1.13) and (1.14) may require minor corrections in cases where several interceptors of a given type are launched from different sites towards a given threat object. These probabilities do in fact depend upon the site location from which the interceptors are launched. Sometimes, a preferred site may be depleted, and may request the assistance of other sites to achieve the total number $m_{\alpha q}$ of interceptors required. If $m_{\alpha qs}$ is the number launched from site s,

$$m_{\alpha q} = \sum_{s=1}^{S} m_{\alpha qs} \quad , \tag{1.15}$$

where S is the total number of sites.

Referring to Equation (1.9), another expression must be substituted for the probability $p_{L_q}(\bullet)$ in accordance with the equality

$$p_{L_q}(m_{\alpha q})\Big|_{O_q=M} = \prod_{s=1}^{S} \left(p_{L_{qs}}(m_{\alpha qs} = 1)\Big|_{O_q=M}\right)^{m_{\alpha qs}} \quad , \tag{1.16}$$

where $p_{L_{qs}}(\cdot)$ is the probability that O_q will leak through if a single interceptor is launched from site s.

But finding the optimal values for $m_{\alpha qs}$ itself requires a multidimensional optimization. To avoid such computations in the calculation of the optimal number $m^*_{\alpha q}$, we assume that interceptors are launched from the most effective site. In these calculations, Equation (1.16) is replaced by the equation

$$p_{L_q}(m_{\alpha q})|_{O_q=M} = \prod_{u=1}^{m_{\alpha q}} \min_{s\in S_{\alpha u}} \left\{ p_{L_{qs}}(m_{\alpha qs}=1)|_{O_q=M} \right\} \quad , \tag{1.17}$$

where $S_{\alpha u}$ is the set of sites at which at least one interceptor of type α is available after the uth interceptor has been allocated (similarly when $O_q = B$).

If, instead of the two-tiered approach where only interceptors of the same type are allowed in any given allocation, we allow a mix of types, then the above minimization can also be extended over types. In such cases, we obtain a probability which is independent of type:

$$p_{L_q}(m_q)|_{O_q=M} = \prod_{u=1}^{m_q} \min_{\substack{s\in S_u \\ \alpha\in\{1,2\}}} \left\{ p_{L_{qs}}(m_{\alpha qs}=1)|_{O_q=M} \right\} \quad . \tag{1.18}$$

The effects of depleting interceptor resources can be studied very easily with this model. Whenever a site runs out of interceptors, the total set of nonempty sites is reduced, and the minimization is bound to yield a larger number. In the extreme, there may not be any interceptors left at all, and $p_{L_q}(m_q)$ must be set to unity. The effects of varying the resources allocated to each site can thus be studied in order to find an appropriate distribution of resources over defense sites.

For such studies, the loss function, Equation (1.13), may be decomposed into two parts, one indicating no depletion and the other total depletion, as follows:

$$L = \sum_{q=1}^{Q_I} \sum_{k=1}^{K} \left[V(a_k) \left(p^M_{qk} p^M_{L_q}(m_{\alpha q}) p^M_q + p^B_{qk} p^B_{L_q}(m_{\alpha q}) p^B_q \right) \right]$$

$$+ \sum_{q=Q_I+1}^{Q} \sum_{k=1}^{K} \left[V(a_k) \left(p^M_{qk} p^M_q + p^B_{qk} p^B_q \right) \right] \quad , \tag{1.19}$$

where O_{Q_I} is the last threat object on the priority list for which a sufficient quantity of interceptors is available.

1.2.2. COMPUTING DEFENSE COSTS

In our simplified models, the defense cost variable C_q of Equation (1.1) represents only the intrinsic ("lifecycle") costs of the interceptors, and has the simple form

$$C_q = m_{\alpha q} C(I_\alpha) \quad , \tag{1.20}$$

where $C(I_\alpha)$ is the lifecycle cost of one interceptor of type α.

An earlier version of our model included in C_q various "loss of protection" costs resulting from the depletion of interceptors. These costs have now been included in the overall risk function of Equation (1.19).

As a concluding comment, Equation (1.13) is a convenient form for L because we can minimize L by maximizing each p_{K_q}, $q = 1, \ldots, Q$ (other things held fixed). The performance of the defense in engaging Q threats can then be judged by how well each individual threat is engaged, and that is why we have emphasized the one-on-one situation in the remainder of this report.

1.3. Threat Prioritization

To avoid the saturation of its sensors, radars, and computers when the defense is confronting massive MaRV attacks, the battle manager processes and updates its information only for a *critical* subset of threat objects. This subset is determined by an appropriate balance between the update rate (clock time increment) and the importance of threats. Three criteria are combined into a threat ranking rule:

1. severity,
2. vulnerability,
3. urgency.

1.3.1. THREAT SEVERITY

Used to reflect the damage expected from the leakage of a threat object if it leaks through, this criterion is simply the conditional expected asset loss L_q derived earlier (Equations (1.5), (1.6), (1.13)):

$$L_q = \sum_{k=1}^{K} V(a_k) \left[p_{qk}^M p_q^M + p_{qk}^B p_q^B \right] \quad . \tag{1.21}$$

1.3.2. THREAT VULNERABILITY

There is no point in launching an interceptor if its p_K is zero, and we include in our ranking rule the probability p^0_{Kq} that O_q will be killed if an interceptor is launched without delay, *i.e.* when $t_{\text{wait}} = 0$. Target O_q is thus ranked lower than target O_r if

$$p_{Kq}(t_{\text{wait}} = 0) < p_{Kr}(t_{\text{wait}} = 0) \quad . \tag{1.22}$$

As usual, p^0_{Kq} depends on the type of O_q, and an expected value is used:

$$p^0_{Kq} = \left(p^0_{Kq}|_{O_q=M}\right) p^M_q + \left(p^0_{Kq}|_{O_q=B}\right) p^B_q \quad . \tag{1.23}$$

The remaining chapters in this report describe how p^0_{Kq} is predicted.

1.3.3. THREAT URGENCY

When an interceptor launch is delayed too long, the interceptor (I) may fail to intercept its assigned threat object (O_q), or it may intercept it inside the keep-out zone, which is not allowed. The maximum amount of waiting time allowed is thus a good measure of the urgency of a threat, and we use that measure in our ranking criterion. This waiting time (τ_q) is simply the difference between the time required by the threat object to reach the keep-out zone and the time required by the interceptor to reach the keep-out zone, at the point X_{ko}. With minor modifications due to the differences between the predicted impact point and the predicted keep-out pierce point, we can use the results of Section 2.2.1.1 to obtain τ_q.

If t^M_{ko} (t^I_{ko}) are the times required by O_q (I) to reach X_{ko}, then

$$\tau_q = t^M_{ko} - t^I_{ko} \quad . \tag{1.24}$$

Using Equation (2.58) to solve for the interceptor flight time t^I_{ko}, and Equations (2.10)–(2.13) with the required modifications to account for the keep-out zone, we obtain

$$t^M_{ko} = \max\{0, \tau_0\} + \min\{t^F_0, t^F\} \quad , \tag{1.25}$$

where

$$\tau_0 = \frac{H_0 - X^0_q \cdot e_y}{e_{V^0_q} \cdot e_y} \quad , \tag{1.26}$$

$$t^F_0 = \frac{-2(H_0 - h_{ko})}{(|V^0_q| + |V^F_q|) e_{V^0_q} \cdot e_y} \quad , \tag{1.27}$$

$$t^F = \frac{-2(X^0_q \cdot e_y - h_{ko})}{(|V^0_q| + |V^F_q|) e_{V^0_q} \cdot e_y} \quad , \tag{1.28}$$

h_{ko} is the height of the keep-out zone,

H_0 is the nominal height of the atmosphere,

X_q^0 is the current position of O_q,

V_q^0 is the current velocity of O_q,

e_y is a unit vector along the altitude coordinate y,

$e_{V_q^0}$ is a unit vector along the current velocity vector V_q^0,

V_q^F is the terminal velocity of O_q.

These results will be derived and explained in Section 2.2.1.1.

1.3.4. OVERALL THREAT RANKING CRITERION

To obtain the overall ranking rule, the three criteria derived above are heuristically combined into the simple ratio

$$r_q = \frac{L_q p_{Kq}^0}{\tau_q} \quad . \tag{1.29}$$

Observe that the optimal probability p_{Kq}^* (and also p_{Kq}^0) depends on the location of the interceptor. Since interceptors can be launched from several sites, we account for this dependence by defining p_{Kq}^* as the maximum probability over all potential launch sites, as derived in Equation (1.17).

1.4. Interceptor Allocation

The principal task in allocating interceptors is to determine the optimal number $m_{\alpha q}^*$ of type α interceptors to be assigned to the highest-priority threat object O_q. As discussed earlier, we approach this problem as a two-tier optimization problem where only two types of interceptors are used: an *overlay interceptor* ($\alpha = 1$) and an *underlay interceptor* ($\alpha = 2$). Overlays are typically used at altitudes exceeding 100,000 feet, where passive maneuvering is seriously limited, and where a thrusting interceptor may have a significant advantage. Underlays are normally used in a "last ditch" defense at lower altitudes where maneuvering capabilities are essential. Such interceptors typically have a shorter range, but a higher acceleration capability, and our example interceptors in Chapter 2 reflect such differences. We do not allow mixed launches in our current version of the battle manager: $m_{\alpha q}^*$ always equals the total number of interceptors allocated to O_q in any given allocation. Threats may reappear on the priority list if they were not killed by the $m_{\alpha q}^*$ interceptors allocated in a previous cycle, in which case a new allocation is made, typically with Type 2 interceptors. To find $m^*_{\alpha q}$, we use an objective function which combines Equations (1.13), (1.17), (1.19), and (1.20):

$$L = \sum_{q=1}^{Q} \left[\sum_{k=1}^{K} \left[V(a_k) \left(p^M_{qk} p^M_{L_q}(m_{\alpha q}) p^M_q + p^B_{qk} p^B_{L_q}(m_{\alpha q}) p^B_q \right) \right] + m_{\alpha q} C(I_\alpha) \right]$$

where

$$p^M_{L_q}(m_{\alpha q}) = \prod_{u=1}^{m_{\alpha q}} \min_{s \in S_{\alpha u}} \left\{ p_{L_{qs}} (m_{\alpha q s} = 1) \,|_{O_q = M} \right\} \quad , \tag{1.30}$$

the minimum leakage probability achievable over all nonempty sites.

As indicated at the outset, we avoid the Q-dimensional optimization by minimizing L over $m_{\alpha q}$ only for $q = 1$, *i.e.*, only for the highest-priority object O_q. The other values $m_{\alpha q}$, $q = 2, \ldots, Q$, are set to 2 for MaRVs and to one for BRVs. This is only a test level for an initial implementation, and it can obviously be refined if experiments indicate it should be done. In the worst case, no such rule can be used, and we are forced to return to the multidimensional approach.

Referring to the sketch of Figure 1.3, the current approach is to find the minimum of the one-dimensional function $F_q = E_q + C_q + W$, where E_q is the expectation of the conditional loss L_q of Equation (1.21), and $q = 1$ since we are considering only the highest priority target. This function is simply Equation (1.30) with all $m_{\alpha q}$ values fixed, except for $m_{\alpha 1}$, and can be expressed as

$$F_1(m_{\alpha 1}) = \sum_{k=1}^{K} V(a_k) \left(p^M_{1k} p^M_{L1}(m_{\alpha 1}) p^M_1 + p^B_{1k} p^B_{L1}(m_{\alpha 1}) p^B_1 \right) + m_{\alpha 1} C(I_\alpha) + W \quad , \tag{1.31}$$

where W is the (fixed) risk associated with the remaining objects $O_2, \ldots, O_Q$. There is no closed form solution for the minimum of this function, and numerical methods must be employed. We use Brent's algorithm (Reference [10]) in our Appendix E to find this minimum. (This algorithm will also be used in Chapter 3 to compute interceptor–MaRV miss distance).

With one simplification, however, a closed form solution to this minimization can be found, and we derived such a solution for exploratory purposes. If we assume, for instance, that $p_{L1}(m_{\alpha 1})$ is site-independent, then $p^M_{L1}(m_{\alpha 1}) = \left(p_{L1}(m_{\alpha 1} = 1)|_{O_1 = M} \right)^{m_{\alpha 1}}$, and similarly for $O_1 = B$.

If we let w be the number of interceptors left when $m_{\alpha 1} = 0$, Equation (1.31) can then be written as

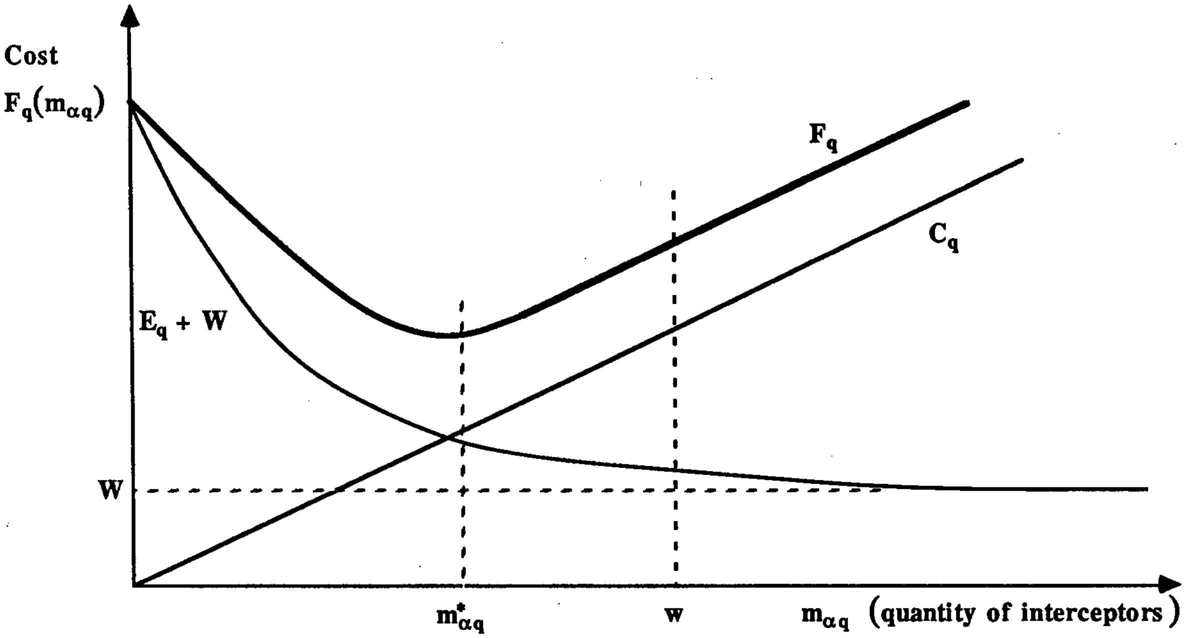


Figure 1.3. The optimal quantity $m^*_{\alpha q}$ of interceptors allocated to the highest-priority threat O_q is found by balancing leakage costs against defense costs subject to the constraint that only w interceptors are available.

$$F_1(m_{\alpha 1}) = \sum_{k=1}^{K} V(a_k) \Big[p_{1k}^M p_1^M \left(p_{L1}(m_{\alpha 1} = 1)|_{O_1 = M} \right)^{\min\{w, m_{\alpha 1}\}}$$

$$+ p_{1k}^B p_1^B \left(p_{L1}(m_{\alpha 1} = 1)|_{O_1 = B} \right)^{\min\{w, m_{\alpha 1}\}} \Big]$$

$$+ \min\{w,\, m_{\alpha 1}\} C(I_\alpha) + W \quad . \qquad (1.32)$$

To find $m^*_{\alpha 1}$, we find the minimum $m^{*0}_{\alpha 1}$ of Equation (1.32), neglecting interceptor depletion constraints. Then we introduce this constraint by setting $m^*_{\alpha 1} = \min\{w,\, m^0_{\alpha 1}\}$. To this end, $F_1(m_{\alpha 1})$ has the form

$$F_1(m_{\alpha 1}) = A(\gamma^M_{\alpha 1})^{m_{\alpha 1}} + B(\gamma^B_{\alpha 1})^{m_{\alpha 1}} + m_{\alpha 1} C(I_\alpha) + W \quad , \qquad (1.33)$$

$$\frac{\partial F_1(m_{\alpha 1})}{\partial (m_{\alpha 1})} = A \ln(\gamma^M_{\alpha 1})(\gamma^M_{\alpha 1})^{m_{\alpha 1}} + B \ln(\gamma^B_{\alpha 1})(\gamma^B_{\alpha 1})^{m_{\alpha 1}} + C(I_\alpha) \quad . \qquad (1.34)$$

If O_q is within radar range, $\gamma^B_{\alpha 1} \approx 0$, and if $m_{\alpha 1} \geq 1$,

$$\frac{\partial F_1(m_{\alpha 1})}{\partial (m_{\alpha 1})} = A \ln(\gamma^M_{\alpha 1})(\gamma^M_{\alpha 1})^{m_{\alpha 1}} + C(I_\alpha) \quad , \tag{1.35}$$

and the solution to $\dfrac{\partial F_1(m_{\alpha 1})}{\partial (m_{\alpha 1})} = 0$ is

$$m^{*0}_{\alpha 1} = \frac{1}{\ln(\gamma^M_{\alpha 1})} \ln\left(\frac{-C(I_\alpha)}{A \ln(\gamma^M_{\alpha 1})}\right) \quad , \tag{1.36}$$

where $A = \sum_{k=1}^K V(a_k) p^M_{1k} p^M_1$. Reintroducing the interceptor attrition constraint, the optimal number of allocated interceptors is then

$$m^*_{\alpha 1} = \min\{m^{*0}_{\alpha 1}, w\} \quad , \tag{1.37}$$

where w is the number of available α-type interceptors.

1.5. Interceptor Launch Logic

Given that the statistical outcome of an engagement is virtually determined by the probability of kill (p_K) of the interceptors, and that this p_K strongly depends on when and where the engagement takes place, launch time and launch direction are dominant parameters in the management of a battle. Premature interceptor launches may cause early burn-out, giving a MaRV considerable speed advantage at the time of intercept. A late launch may not allow the interceptor to reach its maximal velocity prior to or during the engagement, again limiting its effectiveness. Similarly, if an interceptor is launched in an unfavorable direction, a significant energy expenditure may be required to force it on course.

Consequently, and as shown in Figure 1.4, the battle management launch logic subsystem operates on MaRV state information to produce two major outputs:

1. Interceptor launch time,
2. Interceptor launch direction.

These decisions are made and updated at a rate determined by the rate at which the MaRV state updates (X_M) are received from various sensors, and by the internal clock time of the battle manager. Depending upon the quantity of objects on the threat priority list, computer saturation may occur, in which case the decision process must be slowed down accordingly.

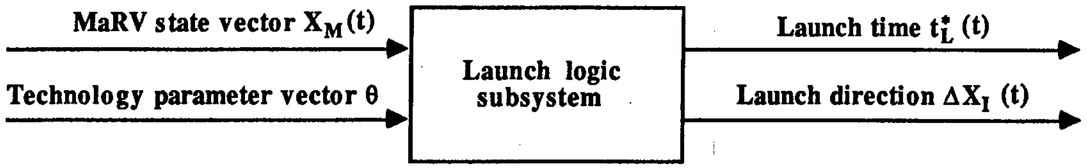

MaRV state vector $X_M(t)$
Technology parameter vector θ
Launch logic subsystem
Launch time t_L^* (t)
Launch direction ΔX_I (t)

Figure 1.4. At each state update time t, the battle management launch logic subsystem operates on the MaRV state vector estimate $X_M(t)$ to produce the optimal launch direction $\Delta^* X_I(t)$ and launch time $t_L^*(t)$.

The output decisions of this logic subsystem also depend upon the technological capabilities and limitations attributed to both the interceptor and the MaRV. Maneuvering, speed, and acceleration capabilities of both vehicles are constrained by limitations in the development and fabrication of metals, composites, and other materials. We represent such constraints in our models by defining limits on weight, lateral acceleration, speed, and other factors, and we represent these as an input vector θ as shown in Figure 1.4.

The optimal launch time (and the optimal waiting time) is found by maximizing p_K over waiting time. Again p_K is the principal variable, with interceptor lethality and miss distance as major parameters:

$$p_{K_q}(m_{\alpha q} = 1) = \text{ prob}\left\{KR_{\alpha q} \geq MD_{\alpha q}\right\} \quad , \tag{1.38}$$

where $KR_{\alpha q}$ is the kill radius of an interceptor of type α against threat O_q,

$MD_{\alpha q}$ is the miss distance achieved by an interceptor of type α against O_q.

In this report, we do not further discuss lethality issues, and we concentrate on the miss distance $MD_{\alpha q}$. The sole purpose of Chapter 3 is to develop an algorithm for predicting $MD_{\alpha q}$, and for optimizing $MD_{\alpha q}$ and $p_{K_{\alpha q}}$ over waiting time.

By extending the MaRV trajectory at the optimal launch time, we predict a point of intercept, and we use this point to determine an interceptor launch direction vector ΔX_I.

1.6. Battle Management Flowchart

A simplified flowchart of the battle manager is shown in Figure 1.5. First, threats are prioritized in accordance with the criterion discussed in Section 1.3., where kill probabilities are maximized over sites and type, as derived in Equation 1.18 (Block 1). Then, the highest-priority threat O_q is placed on another queue in accordance with its t^*_{wq} (3). This second ordering queue is needed because highest-priority threats are not necessarily the most urgent ones, as measured by t^*_{wq}. The overall priority of an object in this t^*_{wq}-queue may vary considerably while it is waiting to be addressed, and only when t^{**}_{wq} approaches the update time increment τ can a final interceptor assignment and launch time be derived. If there is a threat on this waiting queue whose t^*_w is smaller than the update time, continue; otherwise, return to consider next threat (4).

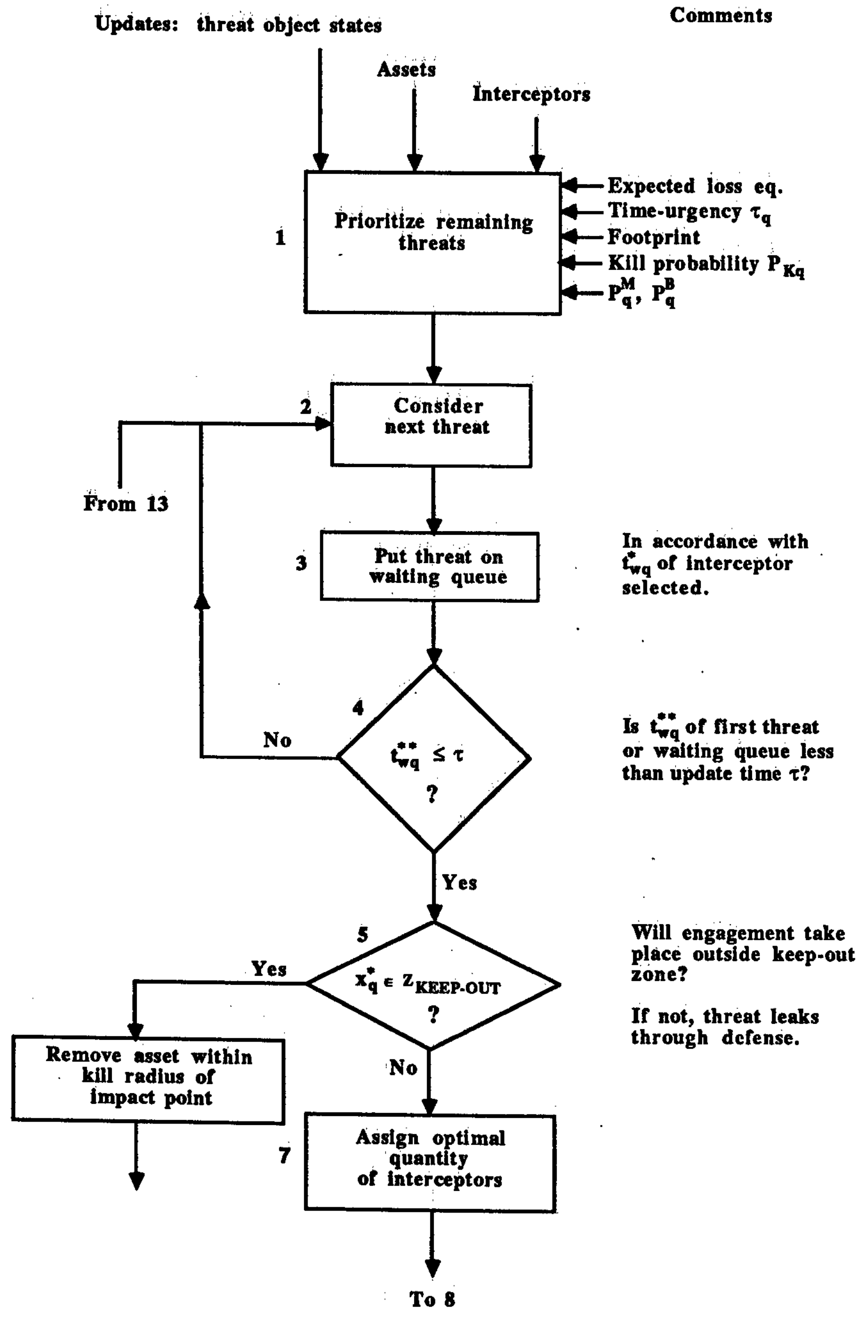
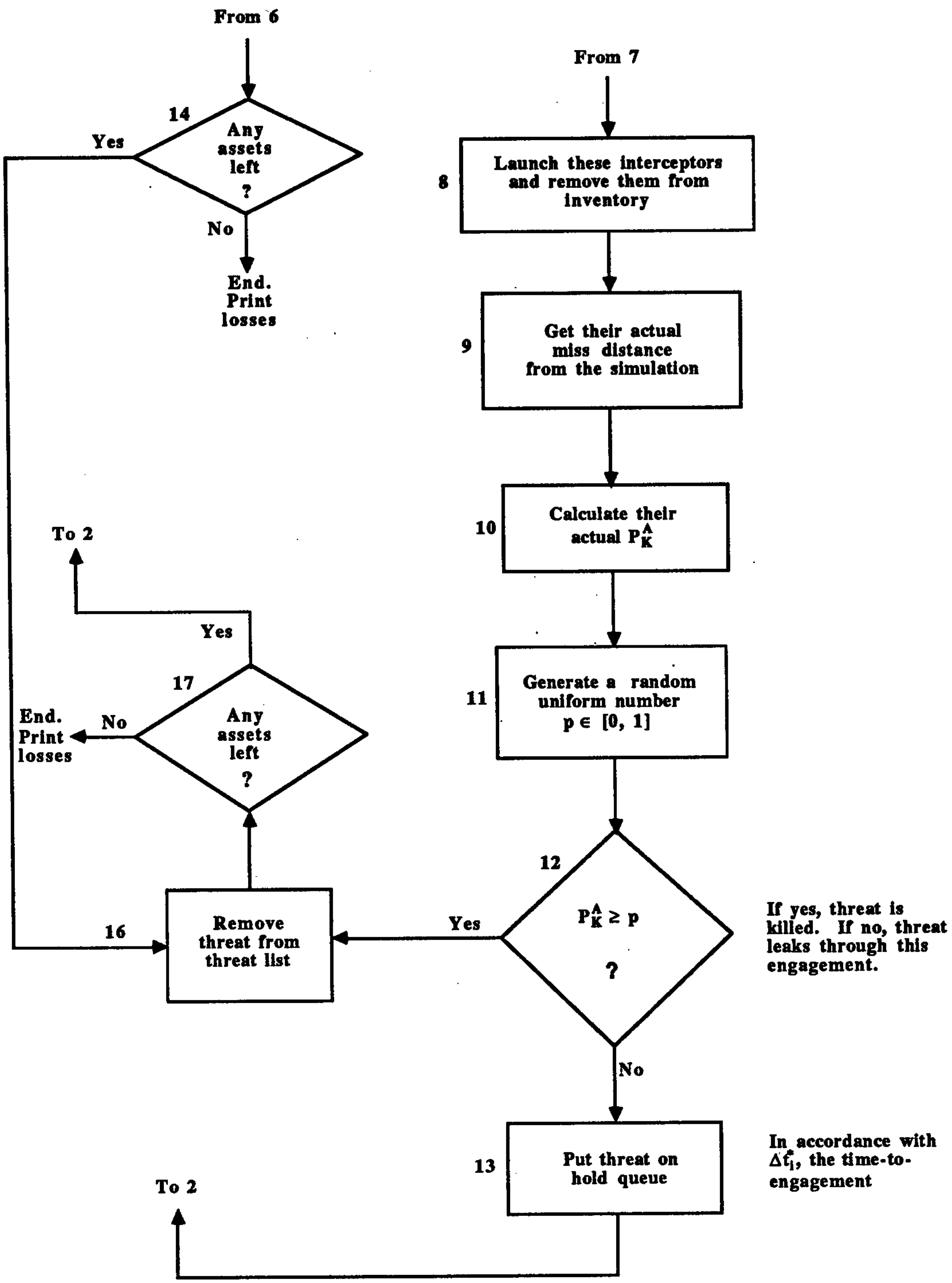

Updates: threat object states
Comments
Assets
Interceptors
1
Prioritize remaining threats
Expected loss eq.
Time-urgency τ_q
Footprint
Kill probability P_{Kq}
P^M_q, P^B_q
2
Consider next threat
From 13
3
Put threat on waiting queue
In accordance with t^*_{wq} of interceptor selected.
4
No
$t^{**}_{wq} \leq \tau$?
Is t^{**}_{wq} of first threat or waiting queue less than update time τ?
Yes
5
Yes
$x^*_q \in z_{KEEP\text{-}OUT}$?
Will engagement take place outside keep-out zone?
If not, threat leaks through defense.
6
Remove asset within kill radius of impact point
No
7
Assign optimal quantity of interceptors
To 8

Figure 1.5. Battle management flowchart (continued on the following page).


From 6
14
Any assets left ?
Yes
No
End. Print losses
From 7
8
Launch these interceptors and remove them from inventory
9
Get their actual miss distance from the simulation
10
Calculate their actual P_K^A
To 2
Yes
17
Any assets left ?
No
End. Print losses
11
Generate a random uniform number $p \in [0, 1]$
12
$P_K^A \geq p$?
If yes, threat is killed. If no, threat leaks through this engagement.
16
Remove threat from threat list
Yes
No
13
Put threat on hold queue
In accordance with Δt_i^*, the time-to-engagement
To 2

If the engagement will take place inside the keep-out zone for the current threat on the waiting list, then the threat leaks through, and the assets within its reach are destroyed (5 and 6). If not, assign the optimal number of interceptors $m^*_{\alpha q}$, launch these inteceptors, and remove them from the inventory (8). Simulate the engagement, obtain the simulated miss distance, and calculate the simulated ("actual") probability of kill p_K^A (9 and 10). Using

p_K^A as the "true" probability, generate a number p from a uniform random distribution (11), and ascertain whether to remove it from the threat list (14). If not, the threat leaks through the current engagement, and must be returned to the active threat list (13), where it is processed as a new threat.

Future chapters will show how some of these computations are done. Chapter 3 in particular will concentrate on the computation of miss distances and waiting times.

Chapter 2

MATHEMATICAL PRELIMINARIES

In this chapter we present the simplified equations describing the motion of MaRVs and interceptors, and we derive reachability and interception conditions needed for the calculation of miss distances. Both vehicles are treated as point masses, and our principal goal is to describe the behavior of these vehicles when they are negotiating simple but stressing maneuvers. First, we discuss general equations applying to both types of vehicles. Then we concentrate on MaRVs and interceptors individually, and we conclude with the derivation of reachability and interception conditions.

2.1. General Equations

A particle of mass m moving with constant speed s and attracted toward a fixed point O by a force F will travel along a circle whose center is at O and whose radius is $r = ms^2/F$.

Consider next a vehicle such as a MaRV or an interceptor experiencing a constant lift force (see Figure 2.1).

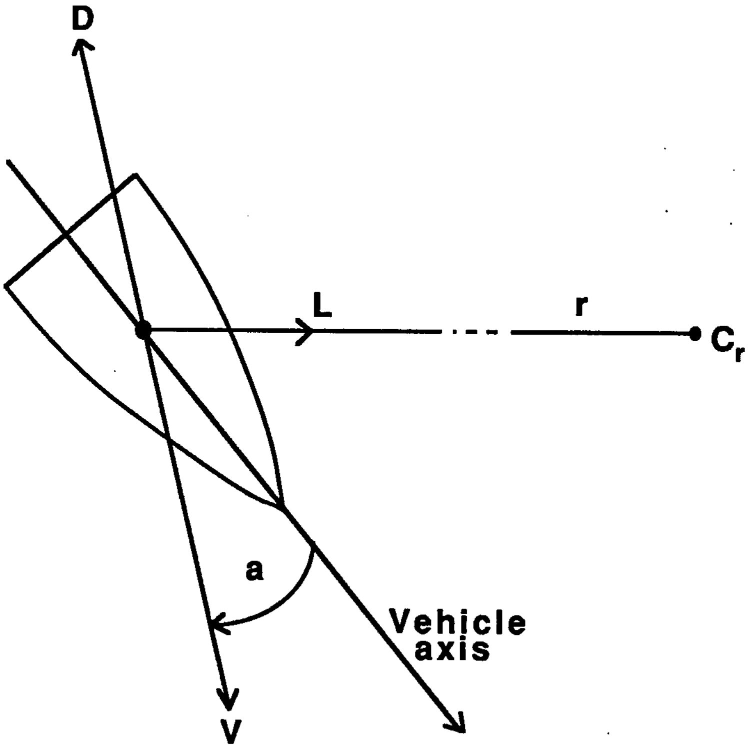


Figure 2.1. Describing the motion of a vehicle under constant lift conditions.

$$L = \frac{1}{2} C_L(\alpha, M)\rho|V|^2 S \quad , \tag{2.1}$$

where:

C_L is the *lift coefficient*,
α is the *angle of attack*,
M is the *Mach number*,
ρ is the *atmospheric density*, $\rho = \rho_0 e^{-h/h_0}$,
$h_0 = 20,000$ fcet (Reference [11], p. F166),
S is the effective area,
$|V|$ is the vehicle speed.

For such a vehicle

$$r = \frac{m|V|^2}{L} = \frac{2m}{C_L(\alpha, M)\rho S} \quad . \tag{2.2}$$

Observe that the speed $|V|$ does not appear in this expression, so that we may ignore the drag

$$D = -\frac{1}{2m} C_D S \rho |V|^2 \quad , \tag{2.3}$$

where C_D is the *drag coefficient.* Admittedly, under some conditions, C_L and M (and occasionally S) are themselves functions of velocity, but such conditions may be neglected for the problem solved in this report. For instance, the effects of Mach number M on the lift coefficient C_L are most pronounced for high values of M (typically $M > 10$), and we will show that optimal evasive engagements rarely take place in such regimes.

There is also the possibility of "induced drag," where a lifting maneuver also causes additional drag, particularly for large values of α. While we are especially interested in maximal maneuvers where α is in fact large, induced drag effects have been shown to be relatively small (a few percent), and we have chosen to neglect these for our simple models. However, such effects can easily be incorporated in our models.

2.2. MaRV Dynamics

In this section, we derive a simplified equation for the speed of a MaRV, and for its radius of curvature under a constant angle-of-attack maneuver.

2.2.1. MaRV SPEED

Above a given height H_0, we assume that drag may be neglected, and the MaRV travels at a constant speed $|V_M^0|$. Below H_0, its speed decreases linearly in time until a terminal (final) speed $|V_M^F|$ at or within the keep-out zone H_{ko} is reached. This linear deceleration behavior is consistent with experimental tests and simulations [12], and is shown in Figure 2.2 below. The vehicle thus coasts at its initial speed $|V_M^0|$ until it reaches the height H_0, beyond which it decelerates at a rate of k_{DM}^{-1} feet/sec^2 until the terminal velocity $|V_M^F|$ is reached. The time taken to reach height H_0 is simply

$$\tau_0 = \frac{H_0 - X_M^0 \cdot e_y}{V_M^0 \cdot e_y} \quad , \tag{2.4}$$

where X_M^0 is the initial position of M (at $t = 0$) and V_M^0 is its initial velocity. We assume that V_M is constant in the interval $[0,\ \tau_0]$.

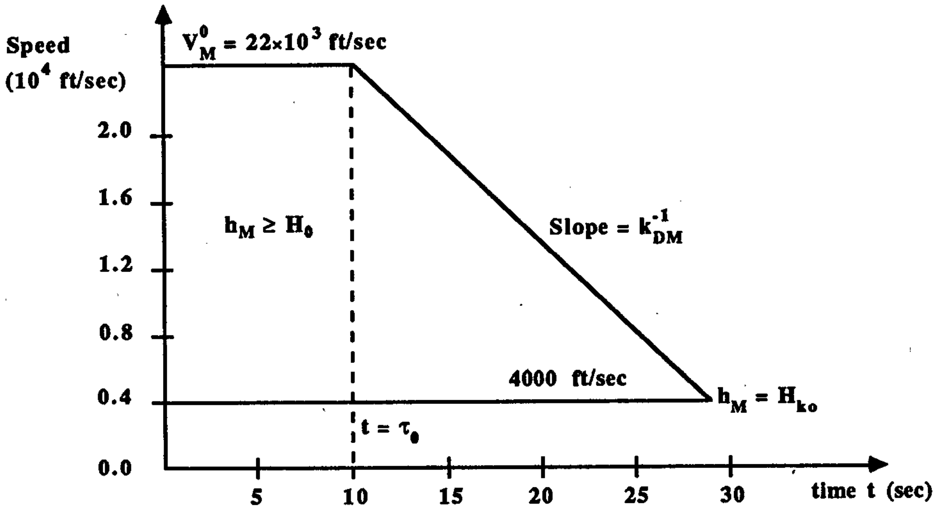


Figure 2.2. Simplified speed profile of a typical MaRV.

The speed profile of M thus has several segments, depending on whether the last speed measurement is made prior to reaching H_0, or afterwards.

$$|V_M(\tau)| = \begin{cases} |V_M^0| \quad , & 0 \leq t \leq \tau_0 \\ |V_M^0| \left[1 - \dfrac{(t - \tau_0)}{k_{DM}}\right] \quad , & 0 \leq \tau_0 \leq t \\ |V_M^0| \left[1 - \dfrac{t}{k_{DM}}\right] \quad , & \tau_0 < 0 \leq t \end{cases} \quad . \tag{2.5}$$

2.2.1.1. The Deceleration Constant k_{DM}

The rate of deceleration (k_{DM}^{-1}) obviously depends upon the initial and final positions and velocities of the MaRV, and upon the height H_0 where drag is assumed to start. To find k_{DM}, we must first determine how long M takes to reach impact (or some other reference point inside the keep-out zone), then we must satisfy the initial and final speed constraints. First, we solve for the impact time t^F. Let $e_{V_M^0}$ be a unit vector in the direction of V_M^0 and e_y a unit vector along the y-axis.

For $X_M^0 \cdot e_y > H_0$, $k_{DM} = \infty$. Below H_0,

$$\frac{-X_M^0 \cdot e_y}{e_{V_M^0} \cdot e_y} = \int_0^{t^F} |V_M^0| \left(1 - \frac{\tau}{k_{DM}}\right) d\tau$$

$$= |V_M^0|\tau \Big|_0^{t^F} - \frac{\tau^2 |V_M^0|}{2k_{DM}} \Big|_0^{t^F} = |V_M^0| t^F - \frac{|V_M^0|(t^F)^2}{2k_{DM}} \quad . \tag{2.6}$$

Applying the velocity constraints gives: $|V_M^0|(1 - t^F/k_{DM}) = V_M^F$, the impact (final) speed.

Thus

$$k_{DM} = \frac{t^F |V_M^0|}{|V_M^0| - |V_M^F|} \quad . \tag{2.7}$$

Substituting (2.7) in (2.6),

$$\frac{-X_M^0 \cdot e_y}{e_{V_M^0} \cdot e_y} = t^F \left[|V_M^0| - \frac{|V_M^0| - |V_M^F|}{2} \right] = \frac{t^F}{2}(|V_M^0| + |V_M^F|)$$

$$= \frac{(|V_M^0| - |V_M^F|)\,(|V_M^0| + |V_M^F|)}{2|V_M^0|}\, k_{DM} \quad . \tag{2.8}$$

Hence

$$k_{DM} = \frac{-2|V_M^0| H_0 \min\{H_0,\, X_M^0 \cdot e_y\}}{e_{V_M^0} \cdot e_y \,(|V_M^0| - |V_M^F|)\,(|V_M^0| + |V_M^F|)} \quad . \tag{2.9}$$

Observe that $e_{V_M^0}$ is assumed to point downwards relative to e_y, hence the dot product in the denominator is always negative and k_{DM} always positive.

The flying time from altitude H_0 to final impact is thus

$$t_0^F = \frac{-2H_0}{(|V_M^0| + |V_M^F|)\, e_{V_M^0} \cdot e_y} \quad , \tag{2.10}$$

and the total flying time from X_M^0 to impact when $X_M^0 \cdot e_y \geq H_0$ is $T_0^F = \tau_0 + t_0^F$, where

$$\tau_0 = \frac{H_0 - X_M^0 \cdot e_y}{e_{V_M^0} \cdot e_y} \quad . \tag{2.11}$$

When the initial altitude of the MaRV is below H_0, $X_M^0 \cdot e_y < H_0$ and we have

$$t^F = \frac{-2X_M^0 \cdot e_y}{(|V_M^0| + |V_M^F|)\, e_{V_M^0} \cdot e_y} \quad . \tag{2.12}$$

In general, the total flying time from any starting position X_M^0 is thus

$$T^F = \max\{0,\, \tau_0\} + \min\{t_0^F,\, t^F\} \quad . \tag{2.13}$$

2.2.2. MaRV POSITION

Starting from an initial position X_M^0, the MaRV position $X_M(t)$ at any future time t is

$$X_M(t) = X_M^0 - V_M^0 \left\{ t - \left[\frac{t^2 - 2\max\{0,\, \tau_0\}t}{2k_{DM}} \right] \frac{\max\{t - \tau_0,\, 0\}}{(t - \tau_0)} \right\} \quad . \tag{2.14}$$

2.2.3. MaRV RADIUS OF CURVATURE

Referring to Equations (2.1) and (2.2), the instantaneous radius of curvature of a MaRV is

$$|r_M(t)| = \frac{2m_M \exp\left\{ \frac{X_M(t)\bullet e_y}{h_0} - 1 \right\}}{\rho_0 C_L(\alpha,\, \mathrm{Mach})_M S_M} \quad , \tag{2.15}$$

where $X_M(t)$ is the position of the MaRV at time t.

When banking or turning is considered (see Figures 2.3 and 2.6), a banking or turning angle ϕ_M must be introduced, and

$$|r_M(t,\, \phi_M)| = \frac{2m_M \exp\left\{ \frac{X_M(t)\bullet e_y}{h_0} + \frac{\Delta h_M(t,\, \phi_M)}{h_0} - 1 \right\}}{\rho_0 C_L(\alpha,\, \mathrm{Mach})_M S_M} \quad , \tag{2.16}$$

where the change in height due to turning through an angle ϕ_M in the "action plane," (defined to be the x–y plane of the Action Coordinate System $(x_M^A,\, y_M^A,\, z_M^A)$) is represented by

$$\begin{aligned} \Delta h_M(t,\, \phi_M) = e_y \bullet \Big[\Big(& R_{z_M^A y_M^A x_M^A}(\gamma,\, \beta,\, \theta) |r_M(t)| \Big) \\ & \times \Big(\sin(\phi_M)\, e_{x_M^A} + (1 - \cos(\phi_M))\, e_{y_M^A} \Big) \Big] \quad , \end{aligned} \tag{2.17}$$

where

$$R_{z_M^A y_M^A x_M^A}(\gamma,\, \beta,\, \theta) \overset{\triangle}{=} R_{e_{z_M^A}}(\gamma) \bullet R_{e_{y_M^A}}(\beta) \bullet R_{e_{x_M^A}}(\theta) \quad , \tag{2.18}$$

rotates the MaRV action coordinates $(x_M^A,\, y_M^A,\, z_M^A)$ into the MaRV coordinates $(x_M,\, y_M,\, z_M)$ (the M-system), and $r_M(t) \overset{\triangle}{=} r_M(t,\, \phi_M|_{\phi_M = 0})$ (see Appendix A).

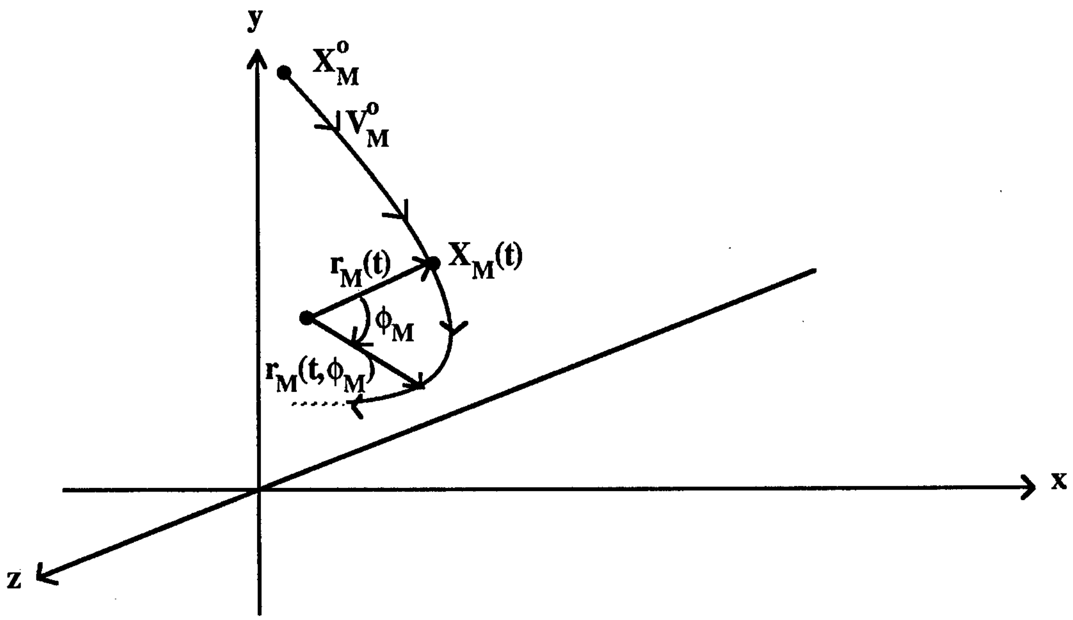


Figure 2.3. The radius of curvature of a vehicle is not only a function of its lateral acceleration, but also of its position in the atmosphere.

When the MaRV maneuver is wholly within its $(x_M - y_M)$ plane, this rotation matrix is evidently the identity matrix, since then $x_M^A = x_M$ and $y_M^A = y_M$.

It should be clear from the above discussion that, although we currently assume that the MaRV maneuver lies wholly within some fixed but otherwise *arbitrary* plane, this assumption can be eliminated by allowing the rotation matrix $R(\gamma, \beta, \theta)$ to vary with time. Any MaRV turn could then be allowed.

2.3. Interceptor Dynamics

The engagement algorithms developed later in this report allow the use of two interceptors: an Overlay Interceptor ("overlay") and an Underlay Interceptor ("underlay").

Overlays are designed to intercept MaRVs early during a battle and at longer distances from their launch points, typically 10^5 feet to 10^6 feet, whereas underlays operate more effectively at shorter distances of about 10^4 to 10^5 feet. The former are more sluggish and tend to require more time to reach their maximum velocity. The latter reach their maximum velocity considerably faster, and are designed for "close-in" engagements.

In this section, we present a summary of the acceleration, velocity, and position profiles of both vehicles. The complete derivation of these profiles is reported in the appendices. We also show the radius of curvature for both vehicles, as we did for the MaRV.

2.3.1. TYPE I INTERCEPTORS: OVERLAYS

These are long-range two-stage interceptors whose weight and thrust profiles are shown in Figures B.1 and B.2 and described by Equations (B1) through (B11) in Appendix B.

2.3.1.1. Thrust Acceleration of Type I Interceptors

Using Equations (B1)–(B11), from Appendix B, the acceleration is

$$a_{I1}(t) = X_1(t)a_{I11}(t) + Y_1(t)a_{I12}(t) + Z_1(t)a_{I13}(t) \quad , \tag{2.19}$$

where

$$X_1(t) = \frac{1 + \mathrm{sign}\,(1.61 - t)}{2} \quad , \tag{2.20}$$

$$Y_1(t) = \left(\frac{1 + \mathrm{sign}\,(3.83 - t)}{2}\right)\left(\frac{1 + \mathrm{sign}\,(t - 1.61)}{2}\right) \quad , \tag{2.21}$$

$$Z_1(t) = \frac{1 + \mathrm{sign}\,(t - 3.83)}{2} \quad , \tag{2.22}$$

$$a_{I11}(t) = \frac{625,500}{230.5(1 - 0.449t) + (m^0_{I1} - 33.59)} \quad , \qquad 0 \leq t \leq 1.61 \quad , \tag{2.23}$$

$$a_{I12}(t) = \frac{114,000}{85.91(1 - 0.159t) + (m^0_{I1} - 33.59)} \quad , \qquad 1.61 \leq t \leq 3.83 \quad , \tag{2.24}$$

and

$$a_{I13}(t) = 0 \quad , \qquad 3.83 \leq t \quad . \tag{2.25}$$

2.3.1.2. Speed of Type I Interceptors

Using Equations (B16)–(B23), the speed is

$$V_{I1}(t) = X_1(t)V_{I11}(t) + Y_1(t)V_{I12}(t) + Z_1(t)V_{I13}(t) \quad , \tag{2.26}$$

where

$$V_{I11}(t) = -6043[\ln(m^0_{I1} + 196.9 - 103.5t) - \ln(m^0_{I1} + 196.9)] \ , \ 0 \leq t \leq 1.61 \ , \tag{2.27}$$

$$\begin{aligned} V_{I12}(t) = {} & 2303\ln(m^0_{I1} + 30.30) + 6043\ln(m^0_{I1} + 196.9) \\ & - 8346\ln(m^0_{I1} + 52.32 - 13.66t) \quad , \qquad 1.61 \leq t \leq 3.83 \quad , \end{aligned} \tag{2.28}$$

$$V_{I13}(t) = V^0_{I13}\left[1 - \frac{t - 3.83}{k_{DI1}}\right] \quad , \qquad 3.83 \leq t \leq 3.83 + k_{DI1} \quad , \tag{2.29}$$

where

$$
\begin{aligned}
k_{DI1} &= 10 \text{ sec} \quad , \\
V^0_{I13} &= V_{I12}(3.83) \\
&= 2303 \ln(m^0_{I1} + 30.30) + 6043 \ln(m^0_{I1} + 196.9) - 8346 \ln(m^0_{I1}) \quad . \qquad (2.30)
\end{aligned}
$$

A typical profile for $m^0_{I1} = 33.59$ is shown in Figure 2.4.

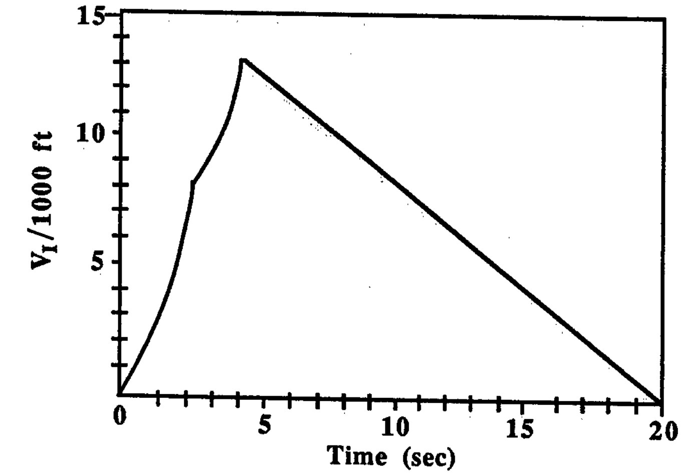


Figure 2.4. Speed profile of a typical Type I interceptor.

2.3.1.3. Position of Type I Interceptors

Using Equations (B24)–(B30), from Appendix B, the distance traveled is (see Figure 2.5)

$$
X_{I1}(t) = X_1(t)X_{I11}(t) + Y_1(t)X_{I12}(t) + Z_1(t)X_{I13}(t) \quad , \qquad (2.31)
$$

where

$$
\begin{aligned}
X_{I11}(t) = 6043 \Big\{ & \left[\ln(m^0_{I1} + 196.9) + 1\right] t \\
& + \left(\frac{m^0_{I1}}{103.5} + 1.90 - t\right) \ln(m^0_{I1} + 196.9 - 103.5t) \\
& - \left(\frac{m^0_{i1} + 196.9}{103.5}\right) \ln(m^0_{I1} + 196.9) \Big\} \ , \ 0 \le t \le 1.61 \quad , \qquad (2.32)
\end{aligned}
$$

$$
\begin{aligned}
X_{I12}(t) = X_{I11}(1.61) &+ [2303 \ln(m_{I1}^0 + 30.30) + 6043 \ln(m_{I1}^0 + 196.9) + 8346]t \\
&+ 8346 \left(\frac{m_{I1}^0}{13.66} + 3.83 - t \right) \ln(m_{I1}^0 + 52.32 - 13.66t) \\
&- (611.0 m_{I1}^0 + 22,238) \ln(m_{I1}^0 + 30.30) + 9729 \ln(m_{I1}^0 + 196.9) + 13,437 \quad , \\
&\qquad 1.61 \le t \le 3.83 \quad , \qquad (2.33)
\end{aligned}
$$

and

$$
\begin{aligned}
X_{I13}(t) = X_{I12}(3.83) + V_{I13}^0 &\left[\left(1 + \frac{3.83}{k_{DI1}} \right) t - \frac{t^2}{2k_{DI1}} - 3.83 \left(1 + \frac{3.83}{2k_{DI1}} \right) \right] \quad , \\
&3.83 \le t \le 3.83 + k_{DI1} \quad . \qquad (2.34)
\end{aligned}
$$

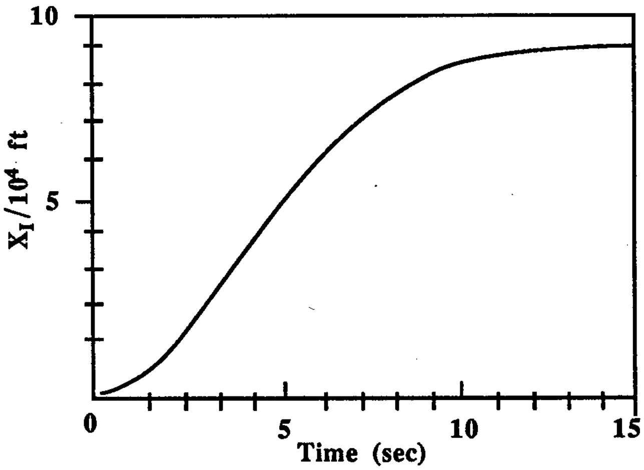


Figure 2.5. Distance traveled for typical Type I Interceptor.

2.3.1.4. Radius of Curvature

Similar to the treatment for MaRVs (see Equations (2.15)–(2.18), the radius of curvature for both interceptors has the same form and depends on where the maneuver is made (determined indirectly by $X_I(\Delta t_I)$), and on the turn angle ϕ_I (see Figure 2.6).

$$
|r_I(\Delta t_I,\, \phi_I)| = \frac{2 m_I(\Delta t_I) \exp \left\{ \dfrac{X_I(\Delta t_I) \bullet e_y}{h_0} + \dfrac{\Delta h_I(\Delta t_I,\, \phi_I)}{h_0} - 1 \right\}}{\rho_0 C_L(\alpha,\, \mathrm{Mach})_I S_I} \quad , \qquad (2.35)
$$

where

$$
\Delta h_I(\Delta t_I,\, \phi_I) = |r_I(\Delta t_I)| \left[(1 - \cos\phi_I)\, e_{z_M} \times e_{V_I^0} + (\sin\phi_I) e_{V_I^0} \right] \bullet e_y \quad , \qquad (2.36)
$$

and θ_c^0 is the collision angle.

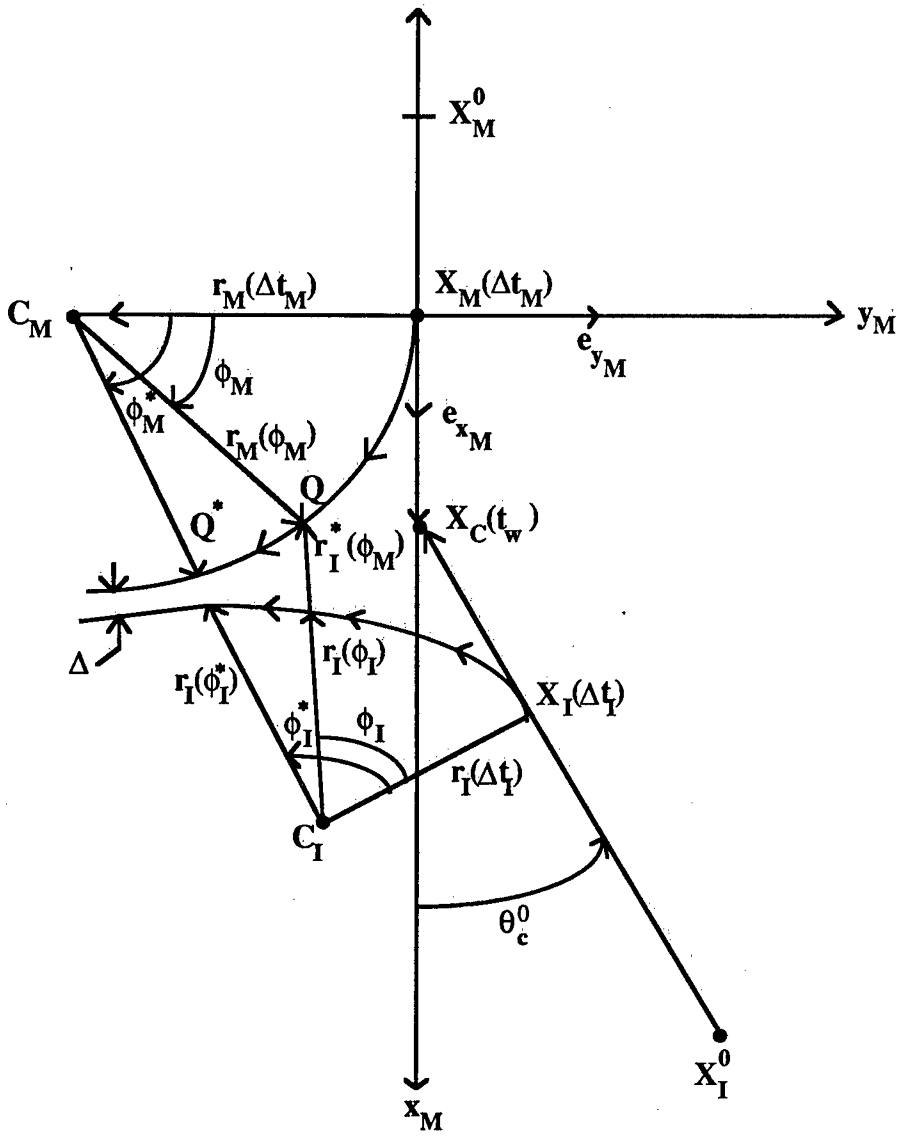


Figure 2.6. Illustrating relative maneuvers of a MaRV and an interceptor.

2.3.2. TYPE II INTERCEPTORS: UNDERLAYS

These are short-range two-stage interceptors whose weight and thrust profiles are shown in Figures C.1 and C.2 of Appendix C.

2.3.2.1. Thrust Acceleration of Type II Interceptors

Using Equations (C1)–(C15) of Appendix C the acceleration is

$$a_{I2}(t) = X_2(t){*}a_{I21}(t) + Y_2(t){*}a_{I22}(t) + Z_2(t){*}a_{I23}(t) \quad , \tag{2.37}$$

where

$$a_{I21}(t) = \frac{2 \times 10^6}{218.8(1 - 0.714t) + m^0_{I2} - 15.62} \quad , \qquad 0 \le t \le 1 \quad , \tag{2.38}$$

$$a_{I22}(t) = \frac{10^5}{109.4(1 - 0.429t) + m^0_{I2} - 15.62} \quad , \qquad 1 \le t \le 2 \quad , \tag{2.39}$$

$$a_{I23}(t) = 0 \quad , \qquad 2 \le t \quad . \tag{2.40}$$

2.3.2.2. Speed of Type II Interceptors

Using Equations (C16)–(C23), the speed is

$$V_{I2}(t) = X_2(t){*}V_{I21}(t) + Y_2(t){*}V_{I22}(t) + Z_2(t){*}V_{I23}(t) \quad , \tag{2.41}$$

where

$$V_{I21}(t) = -12,803[\ln(203.18 + m^0_{I2} - 156.2t) - \ln(203.18 + m^0_{I2})], \ 0 \le t \le 1, \tag{2.42}$$

$$\begin{aligned} V_{I22}(t) = {} & 12,803 \ \ln(203.18 + m^0_{I2}) - 1710 \ \ln(46.98 + m^0_{I2}) \\ & - 2131 \ \ln(93.78 + m^0_{I2} - 46.98t) \ , \ 0 \le t \le 2, \end{aligned} \tag{2.43}$$

$$V_{I23}(t) = V^0_{I23} \left[\left(1 - \frac{(t-2)}{k_{DI2}} \right) \right] \quad , \qquad 2 \le t \le 2 + k_{DI2} \quad , \tag{2.44}$$

where

$$k_{DI2} = 5 \text{ sec} \quad ,$$

$$V^0_{I23} = 12,803 \ \ln(203.18 + m^0_{I2}) - 10,672 \ \ln(46.98 + m^0_{I2}) - 2131 \ \ln(m^0_{I2}) \quad .$$

A typical speed profile is shown in Figure 2.7.

2.3.2.3. Position of Type II Interceptors

Using Equations (C24)–(C31), the distance traveled equals (see Figure 2.8)

$$X_{I2}(t) = X_2(t) * X_{I21}(t) + Y_2(t) * X_{I22}(t) + Z_2(t) * X_{I23}(t) \quad , \tag{2.45}$$

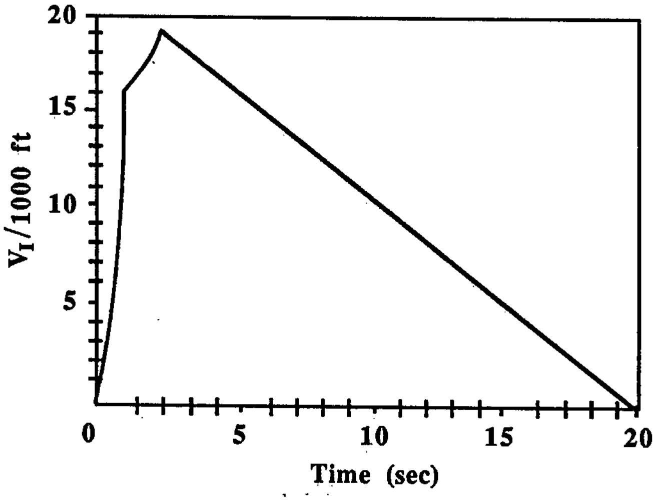


Figure 2.7. Typical speed profile of a Type II interceptor with mass = 15.62 slugs.

where

$$X_{I21}(t) = 12{,}803\Big\{\,(\ln(203.18 + m^0_{I2}) + 1) + 1)\,t - \left(1.3 + \frac{m^0_{I2}}{156.2}\right)\ln(203.18 + m^0_{I2})$$

$$+ \left(1.3 + \frac{m^0_{I2}}{156.2} - t\right)\ln(203.18 + m^0_{I2} - 156.2\,t)\Big\} \quad , \qquad 0 \le t \le 1 \ , \quad (2.46)$$

$$X_{I22}(t) = X_{I21}(1) + \big(12{,}803\,\ln(203.18 + m^0_{I2}) - 106.72\,\ln(46.98 + m^0_{I2}) + 2131\big)\,t$$

$$- 12{,}803\,\ln(203.18 + m^0_{I2}) - 2131 + (8541 - 45.44 m^0_{I2})\ln(46.98 + m^0_{I2})$$

$$+ (4261 + 45.44 m^0_{I2} - 2131\,t)\ln(93.78 + m^0_{I2} - 46.9\,t)\ , \ 1 \le t \le 2\ , \quad (2.47)$$

$$X_{I23}(t) = X_{I22}(2) - V^0_{I23}\left(2 + \frac{2}{k_{DI2}} - \left(1 + \frac{2}{k_{DI2}}\right)t + \frac{t^2}{2k_{DI2}}\right)\ ,$$

$$2 \le t \le 2 + k_{DI2} \quad , \qquad (2.48)$$

where

$$V^0_{I23} = 3841\,\ln(203.18 + m^0_{I2}) - 1710\,\ln(46.98 + m^0_{I2}) - 2131\,\ln(m^0_{I2}).$$

2.3.2.4. Radius of Curvature

The radius of curvature for a Type II has the same form as that for a Type I, with possibly different values for the payload mass m^0_I, the lift coefficient $C_L(\alpha, \text{Mach})_I$, and the effective area S_I.

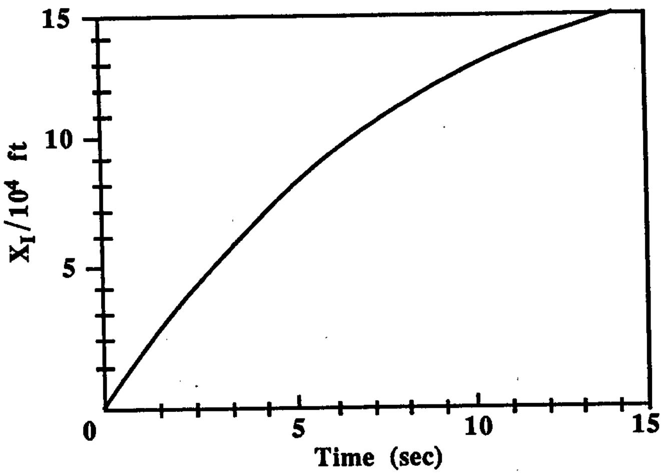


Figure 2.8. Distance traveled for a typical Type II interceptor.

2.4. Computing Intercept Conditions

The magnitude of miss distances depends very strongly on conditions which prevail at the time and location of intercept. In some cases, miss distances may never have to be computed because of intrinsic limitations in either the MaRV or the interceptor. For instance, the MaRV trajectory may not be reachable by a given interceptor, as in the case of a MaRV flying overhead at high altitude and aimed at a distant target. Such cases should be detected early on, to avoid unnecessary computations by the battle manager.

When a MaRV *is* reachable, it is important to know—or at least to estimate—where, when, and at what angle of incidence, the interception will occur. If that angle exceeds $\pi/2$ radians, for instance, a tail-chase condition arises where the engagement is essentially determined by only the relative velocity of both vehicles, and relative maneuvering capabilities become secondary.

In this section, we derive the various intercept conditions which will be important in our miss distance estimation.

2.4.1. MaRV TRAJECTORY REACHABILITY

If the point on the MaRV trajectory nearest to the interceptor launch site is not reachable by the interceptor, then surely no other point on the trajectory will be. Consider Figure 2.9, where X_M^0 and V_M^0 are defined as usual, X_I^0 is the launch position of Interceptor I, and X_C is some intercept ("collision") point along the vector from X_M^0 to X_M^F, the MaRV impact point. For this report, we estimate X_M^F by extending the velocity vector

V_M^0 of M to the $(x,\, y)$-plane. It is easily shown that

$$X_M^F = X_M^0 - \frac{X_M^0 \cdot e_y}{e_{V_M^0} \cdot e_y}\, e_{V_M^0} \quad , \qquad (2.49)$$

when e_y is a unit vector along the y-axis (when $e_{V_M^0} \cdot e_y = 0$, X_M^F obviously does not exist).

At the nearest point, $X_C^{\pi/2}$, ΔX_I and $e_{V_M^0}$ are orthogonal, *i.e.*, their dot product must vanish:

$$\Delta X_I \cdot e_{V_M^0} = 0 \quad . \qquad (2.50)$$

But $\Delta X_I = X_C^{\pi/2} - X_I^0$, and

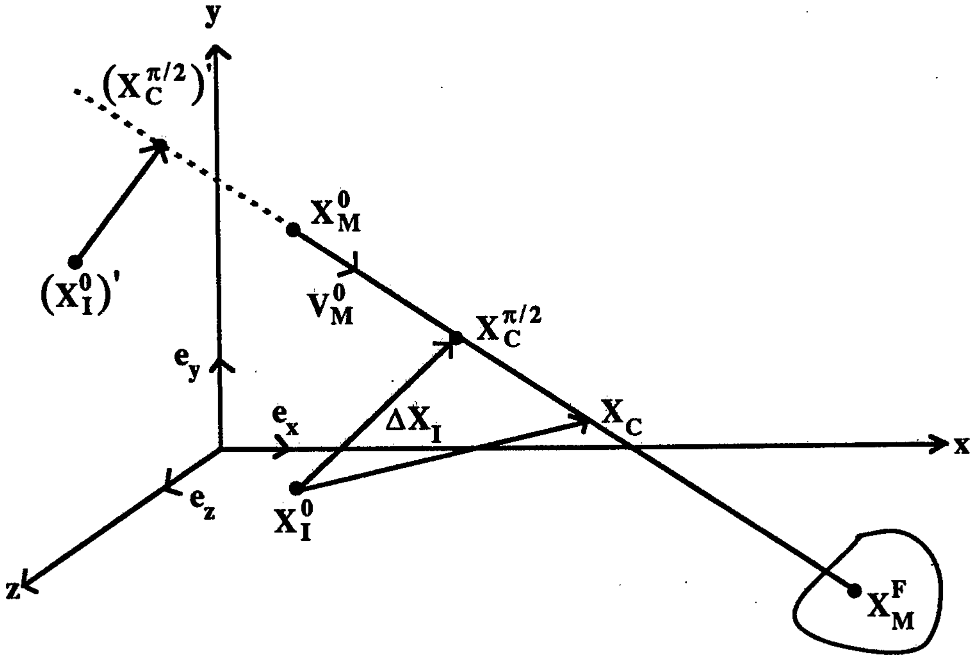


Figure 2.9. Simplified interception geometry illustrated.

$X_C^{\pi/2} = X_M^0 + k_c e_{V_M^0}$. Thus $\Delta X_I = X_M^0 + k_c e_{V_M^0} - X_I^0$.

Hence $(X_M^0 + k_c e_{V_M^0} - X_I^0) \cdot e_{V_M^0} = 0$, or $X_M^0 \cdot e_{V_M^0} + k_c - X_I^0 \cdot e_{V_M^0} = 0$.

Thus

$$k_c = e_{V_M^0} \cdot (X_I^0 - X_M^0) \quad , \qquad (2.51)$$

and

$$X_C^{\pi/2} = X_M^0 + e_{V_M^0} \cdot (X_I^0 - X_M^0) \, e_{V_M^0} \quad . \tag{2.52}$$

Observe that $X_C^{\pi/2}$ need not lie between X_M^0 and X_M^F, as in the case where the interceptor is based at location $(X_I^0)'$. In such cases, k_c is negative and reachability is less likely.

To determine trajectory reachability, we need to solve for the time $t_{IC}^{\pi/2}$ required by the interceptor I to travel from X_I^0 to $X_C^{\pi/2}$. We thus need to solve

$$\Delta X_I = |X_C^{\pi/2} - X_I^0| = \int_0^{t_{IC}^{\pi/2}} |V_I(\tau)| d\tau \qquad \text{for } t_{IC}^{\pi/2} \quad . \tag{2.53}$$

If there is no solution, the MaRV trajectory—*a fortiori* the MaRV itself—is not reachable.

2.4.2. MaRV REACHABILITY AND INTERCEPT PREDICTION

Even if the MaRV trajectory is reachable, the MaRV itself may not be, because the interceptor may be launched at a time when the MaRV is sufficiently close to $X_C^{\pi/2}$ that it will have traveled beyond $X_C^{\pi/2}$ when the interceptor arrives there. In actual situations, of course, the interceptor guidance system will force it to track the MaRV past $X_C^{\pi/2}$, but the MaRV may still no longer be reachable. If it is, a tailchase ensues whose outcome is essentially determined by the relative velocity between the two vehicles.

Observe that the initial intercept estimate X_C, as shown in Figure 2.10, is not only important for reachability purposes, but also to establish initial conditions for problems where interceptor and MaRV maneuvers are considered, as we shall see in later sections. Equally important is the fact that the X_C is used as an aimpoint for the interceptor, which is launched along the vector ΔX_I from X_I^0 to X_C.

Consider thus a MaRV M traveling along a straight line defined by V_M^0 and by the initial position X_M^0, and an interceptor I located or based at point X_I^0, as shown in Figure 2.10.

Let $e_{V_M^0}$ and $e_{X_M^0}$ be unit vectors along the initial velocity V_M^0 and the initial position X_M^0, respectively, d_{MC}^0 be the distance from X_M^0 to the (unknown) intercept point X_C and $d_{CI}^0 = |\Delta X_I|$ and $\Delta X_I \triangleq X_C - X_I^0$. These quantities clearly depend upon the waiting time t_{wait} to launch the interceptor I.

The distance $d_{MV} = |X_M^0 - X_{MV}^F|$ to impact is simply $d_{MV} = |X_M^0 \cdot e_y / e_{V_M^0} \cdot e_y|$. Furthermore,

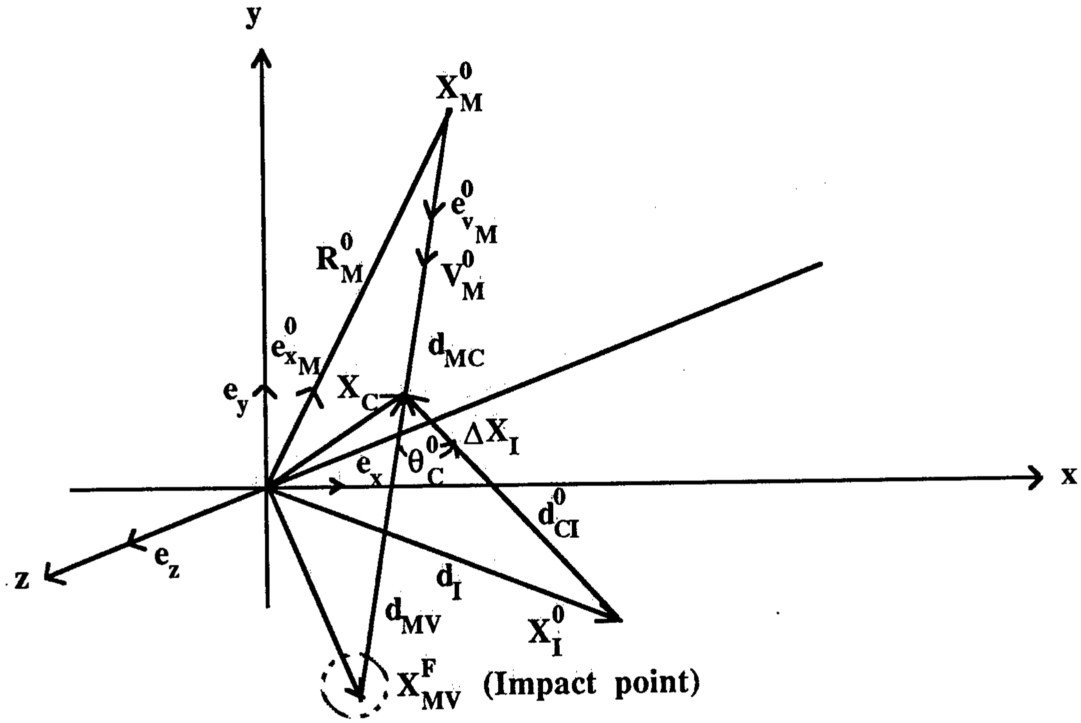


Figure 2.10. Geometric conditions to derive the intercept point.

$$\begin{aligned} X_M^0 + d_{MC}^0(t_{\text{wait}}) e_{V_M^0} &= X_C(t_{\text{wait}}) \\ &= X_I^0 + \Delta X_I(t_{\text{wait}}) \quad , \end{aligned} \tag{2.54}$$

but

$$\Delta X_I(t_{\text{wait}}) = X_M^0 + d_{MC}^0(t_{\text{wait}}) e_{V_M^0} - X_I^0 \quad . \tag{2.55}$$

Hence, if we define $t_{cw} \triangleq t_{IC} + t_{\text{wait}}$, where t_{IC} is the interceptor time-to-collision (total flight time),

$$\int_0^{t_{IC}} |V_I(\tau)| d\tau \; e_{\Delta X_I} = (X_M^0 - X_I^0) + \int_0^{t_{cw}} |V_M(\tau)| d\tau \; e_{V_M^0} \quad , \tag{2.56}$$

where

$$X_M^0 - X_I^0 \triangleq X_{MI}^0 = (X_{MIx}, X_{MIy}, X_{MIz})$$

and

$$e_{V_M^0} = \left(\frac{V_{Mx}^0}{|V_M^0|}, \frac{V_{My}^0}{|V_M^0|}, \frac{V_{Mz}^0}{|V_M^0|} \right) \quad .$$

The right-hand side (RHS) of Equation (2.56) is thus

$$\text{RHS} = \left(X_{MIx}^0 + \int_0^{t_{cw}} \frac{|V_M(\tau)| \, d\tau}{|V_M^0|} V_{Mx}^0 \; , \quad X_{MIy}^0 + \int_0^{t_{cw}} \frac{V_{My}^0}{|V_M^0|} |V_M(\tau)| \, d\tau \; , \right.$$

$$X^0_{MIz} + \int_0^{t_{cw}} \frac{V^0_{Mz}|V_M(\tau)|\,d\tau}{|V^0_M|} \Bigg) \triangleq \left(U_x\left(t_{cw}\right)\ ,\ U_y\left(t_{cw}\right)\ ,\ U_z\left(t_{cw}\right)\right)\ . \tag{2.57}$$

But the magnitude of RHS and the left-hand side (LHS) must be equal. Hence

$$\int_0^{t_{IC}} |V_I(\tau)|\,d\tau = \left[U_x(t_{cw})^2 + U_y(t_{cw})^2 + U_z(t_{cw})^2\right]^{1/2} \quad . \tag{2.58}$$

To determine the time t_{IC} elapsed from interceptor launch to collision or interception, it thus suffices to solve Equation (2.58) for t_{IC}.

To find the root t_{IC} of Equation (2.58), we use algorithm RTSAFE [10]. This algorithm requires the first derivatives of the functions in Equation (2.58), and these are derived in Appendix D.

Once t_{IC}—and hence t_{cw}—are known, d_{MC} and d^0_{CI} can readily be found since

$$d^0_{MC} = \int_0^{t_{cw}} |V_M(\tau)|\,d\tau \tag{2.59}$$

and

$$d^0_{CI} = \int_0^{t_{IC}} |V_I(\tau)|\,d\tau \quad . \tag{2.60}$$

We can now also compute the intercept angle θ^0_c, which is needed to compute miss distance and to determine the direction of interceptor launch.

To solve for θ^0_c, consider that

$$X_C(t_{\text{wait}}) = X^0_M - d^0_{MC}(t_{\text{wait}})e_{V^0_M} \tag{2.61}$$

and

$$\Delta X_I(t_{\text{wait}}) = X_C(t_{\text{wait}}) - X^0_I \quad . \tag{2.62}$$

Hence,

$$\begin{aligned}\cos(\theta^0_c) &= \frac{-\Delta X_I(t_{\text{wait}})\cdot e_{V^0_M}}{|\Delta X_I|} = \frac{-\Delta X_I(t_{\text{wait}})\cdot e_{V^0_M}}{\int_0^{t_{IC}} |V_I(\tau)|\,d\tau} \\ &= \frac{-(X^0_M + d^0_{MC}(t_{\text{wait}})e_{V^0_M} - X^0_I)\cdot e_{V^0_M}}{\int_0^{t_{IC}} |V_I(\tau)|\,d\tau} \\ &= \frac{-(X^0_M + X^0_I)\cdot e_{V^0_M} - d^0_{MC}(t_{\text{wait}})e_{V^0_M}}{\int_0^{t_{IC}} |V_I(\tau)|\,d\tau} \quad ,\end{aligned} \tag{2.63}$$

where $e_{\Delta X_I} \equiv e_{V^0_I}$, a unit vector along the interceptor launch direction.

Chapter 3

PREDICTING MISS DISTANCES

In Chapter 1, we presented a summary of the battle management problem motivating this report. In this chapter, we narrow our focus to the engagement of a single MaRV by a single interceptor, as shown in Figure 3.1. Simple laws for the behavior of both vehicles were derived in the preceding chapter, together with various MaRV reachability and interception conditions. First, we provide a general overview of this one-on-one problem. Then we construct the miss distance prediction algorithm upon which the battle manager is based, and we discuss each of its pieces in detail.

In the typical scenario illustrated in Figure 3.1, a MaRV is headed towards some target defended by a collection of interceptors. In order to decouple our method from major tactical assumptions, we do not make the currently popular assumption that the MaRV will execute its evasive or distracting maneuvers at times or altitudes predetermined by some internal computer program, although our method accommodates such an assumption as a special case. This way, we also allow advanced MaRV systems which can estimate the state of the approaching interceptor throughout the engagement, either with an onboard sensor, or from some external source.

After acquisition by the Search and Acquisition Sensor (SAS) is complete, the Acquisition and Tracking Radar (ATR) tracks the MaRV towards its destination. At some optimal time $t = t_L$ the interceptor is launched towards the MaRV, guided by the Guidance Radar (GR) which uses tracking data received from the ATR as the engagement proceeds. This optimal time t_L is the time at which the predicted probability of kill is a maximum, and it is determined by the predicted miss distance, engagement altitude, and interceptor warhead lethality. A capability to predict the probability of kill achievable by an interceptor (the p_k of the interceptor) is thus essential to effective battle management. Recall from Chapter 1 that this probability involves interceptor lethality and interceptor-MaRV miss distance as major parameters:

$$p_{k_q}(m_{\alpha q} = 1) = \text{prob}\{KR_{\alpha q} \geq MD_{\alpha q}\} \quad , \tag{3.1}$$

where $KR_{\alpha q}$ is the kill radius of an interceptor of type α against threat O_q,
$MD_{\alpha q}$ is the miss distance achieved by an interceptor of type α against O_q.

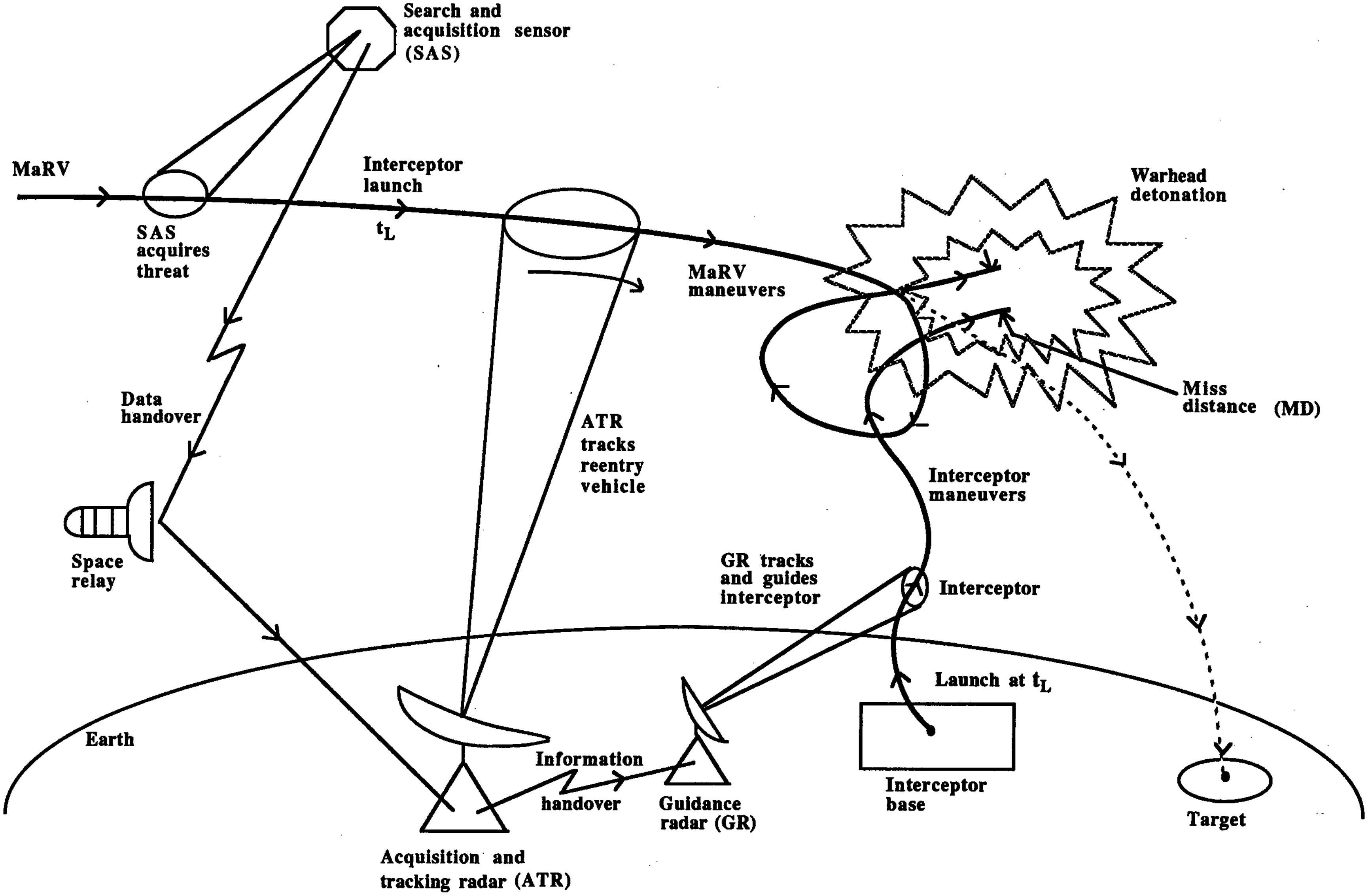


Figure 3.1. Major elements of a one-on-one engagement.

In this report, we omit further discussion of lethality issues, and instead concentrate on the miss distance $MD_{\alpha q}$. In particular, the purpose of this chapter is to develop an algorithm for predicting and minimizing $MD_{\alpha q}$, and we call this algorithm the MD algorithm.

The miss distance achieved by an interceptor is determined by the dynamical state of both vehicles at the time of engagement—their position, velocity, and acceleration—and by their maneuverability. Because atmospheric density is an exponential function of altitude, and the state of both vehicles varies significantly with time, vehicle dynamics and maneuverability are sensitive functions of *when* and *where* an engagement takes place. This affords the battle manager considerable control over the outcome of an engagement since, by controlling the launch time of an interceptor, the battle manager strongly influences the time and place of the engagement, hence the miss distance and the p_k. Besides obvious resource parameters like interceptor quantity and type, launch waiting time (t_w) is in fact the most important parameter under the battle manager's control.

Interceptor performance against a MaRV is also sensitive to the quality of guidance information and commands received from the guidance radar and from various sensors. Not only are MaRV state estimates corrupted by several types of noise, but these estimates invariably reach the interceptor with some delay. This constrains the interceptor to base its decisions on partially erroneous and obsolete data, thereby increasing its miss distance. The miss distance prediction algorithm developed in this report accounts for both sources of error. The effects of interceptor system delays are discussed in Section 3.4.2, and a model for uncertainty and error propagation is presented in Section 3.4.3.

As shown in Figure 3.2, the Miss Distance (MD) algorithm produces three principal outputs in response to an input X_M from the MaRV tracker:

1. Opportunity windows.
2. Miss Distance for an undelayed launch (MD^0).
3. Launch direction vector (ΔX_I).

As the launch waiting or delay time t_w is varied, the miss distance function $MD(t_w)$ will typically exhibit more than one minimum for a given MaRV state estimate X_M. Regions where such minima are attained are called *opportunity windows*, and these are the most important outputs of MD. As we shall see later in this chapter, such windows are not necessarily connected subsets of the waiting time scale, and may range in size from singletons to rather broad intervals.

The particular miss distance MD^0 obtained when the interceptor is launched without delay ($t_w = 0$) is important in selecting interceptors because it affects the prioritization of targets, as we have seen in Chapter 1. We also have a rather impatient battle manager that prefers launching too early over launching too late, and it uses MD^0 to select the exact launch time.

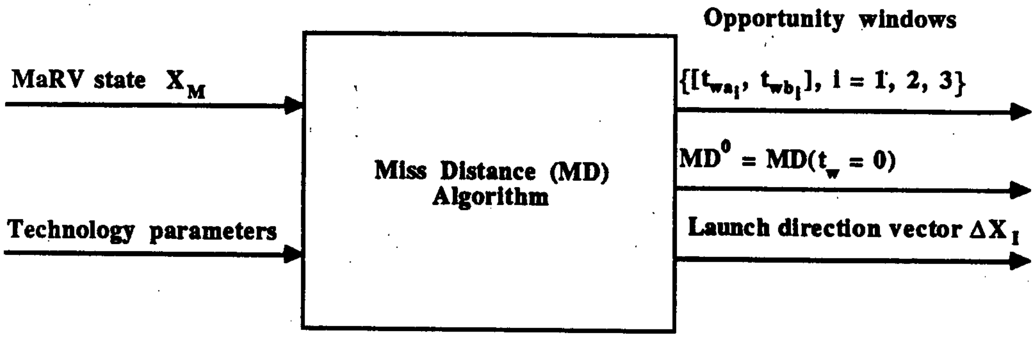

MaRV state X_M
Technology parameters
Miss Distance (MD) Algorithm
Opportunity windows
$\{[t_{wa_i}, t_{wb_i}], i = 1, 2, 3\}$
$MD^0 = MD(t_w = 0)$
Launch direction vector ΔX_I

Figure 3.2. For each MaRV state update X_M, algorithm MD estimates opportunity windows, miss distances, and interceptor launch direction.

The vector ΔX_I pointing from the interceptor launch site X_I^0 towards the predicted intercept point X_c is called the *launch vector* and is used as a launch direction by the battle manager.

To produce the three outputs shown in Figure 3.2, the miss distance algorithm analyzes an engagement in terms of four regions, as illustrated by the typical geometry of Figure 3.3, where H_0 is the idealized boundary between the exoatmosphere and the endoatmosphere, and H_{ko} is the height of the keep-out zone:

1. The Extremal Region (R_0).
2. The Exoatmospheric Region (R_1).
3. The Intermediate Region (R_2).
4. The Tailchase Region (R_3).

For each new MaRV position update X_M^0 and velocity update V_M^0 from the ATR, these four regions are re-evaluated to update the opportunity windows, assuming that the MaRV travels along a straight line determined by X_M^0 and V_M^0. This straight line continuation is a good assumption because, even though the MaRV will almost certainly maneuver in the future, the battle manager has little knowledge about such maneuvers. Assumptions about accelerations and other higher-order terms are considerably less robust since such terms are subject to more rapid and less predictable changes. Furthermore, as launch time approaches, the time-to-go decreases, and the possible error between future MaRV velocities and their most recent tracker update diminishes rapidly.

The *extremal region* denotes combinations of geometric and dynamic conditions where the MaRV is beyond reach of the interceptor, regardless of the value of t_w. Consider for instance an interceptor I based at X_I^0, and a MaRV M traveling from X_M^0 to X_M^F along a straight line trajectory, as shown. Then M is unreachable by I, regardless of the waiting time t_w, if the point $X_c^{\pi/2}$ on the trajectory closest to X_I^0 is not reachable by I.

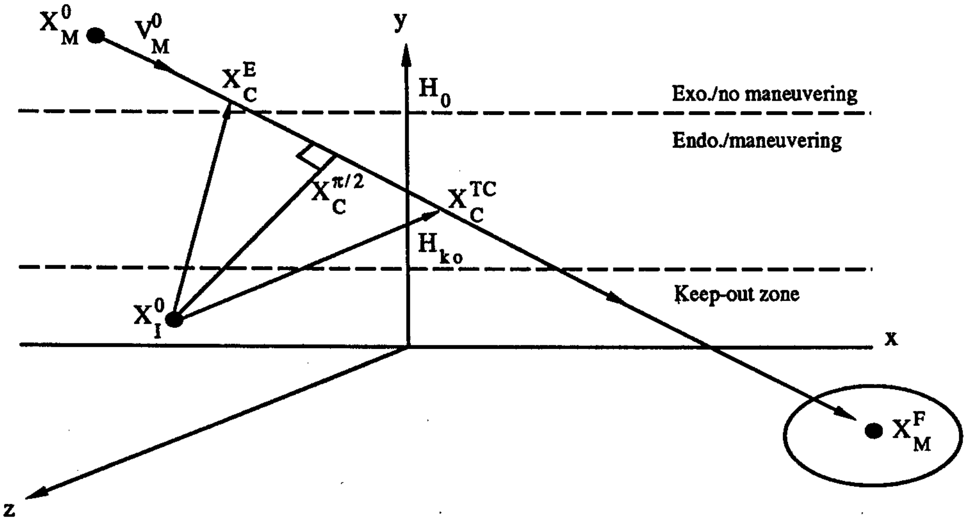


Figure 3.3. A simplified interception geometry used in the computation of reachability and interception regions.

The *exoatmospheric region* is defined in terms of engagements which take place at altitudes above H_0 (typically 100,000 feet). At these heights, MaRV maneuverability is severely limited, and the miss distance is determined by the reach of the interceptor. If, for some waiting time t_w, there exists a point $X_c^E(t_w)$ in that region which is simultaneously reachable by both vehicles—*i.e.*, $X_c^E(t_w)$ is a potential interception point—and if the velocity of the interceptor arriving at the interception point exceeds zero, we set the miss distance to zero for this value of t_w, and t_w is said to belong to the exoatmospheric region. We thus assume that M behaves ballistically above H_0 and that I, if given perfect information about the position and velocity of M, will intercept M without error if M is reachable for t_w. The velocity constraint on I at X_c^E is necessary to eliminate the unreasonable possibility that X_c^E be "parked" at X_c^E until M reaches it.

The *tailchase region* is determined by the relative velocities of the MaRV and the interceptor. If the predicted intercept or collision point for a given t_w is $X_c^{TC}(t_w)$, and if the dot product of their velocities at $X_c^{TC}(t_w)$ is positive, we say that t_w belongs to the tailchase region. If the MaRV speed at the collision point exceeds that of the interceptor, the MaRV is assumed to win the tailchase, and the miss distance is set to zero. If not, it is set to some very large number. This condition for a hit is a bit stronger than that typically found in the literature (see [13], for instance), but it is simpler, and appropriate for our problem.

A waiting time t_w belongs to the *intermediate region* if it does not belong to any other region. It is the only region where evasive maneuvering has a significant effect on miss distance, and it is thus also the most difficult to analyze since the miss distance algorithm must carefully account for the relative maneuvering capabilities of both vehicles. As we describe below, the miss distance estimates in this region are based on a simple but revealing test consisting of a single maximal evasive turn by the MaRV, which is countered by a maximal turn from the interceptor.

Summarizing this introductory discussion, the three regions R_1, R_2, and R_3 are subsets of waiting times t_w, and the MD algorithm produces three *windows* $W_1 = [t_{w11}, t_{w12}] \subset R_1$, $W_2 = [t_{w21}, t_{w22}] \subset R_2$, and $W_3 = [t_{w31}, t_{w32}] \subset R_3$, as shown in Figure 3.4. Each window $W_i \subset R_i$ is an interval of t_w values for which the miss distance is a minimum in region R_i, $i = 1, 2, 3$. Since R_0 is the extremal region where the miss distance is arbitrarily large, it has no window, and no information about R_0 is handed to the battle manager.

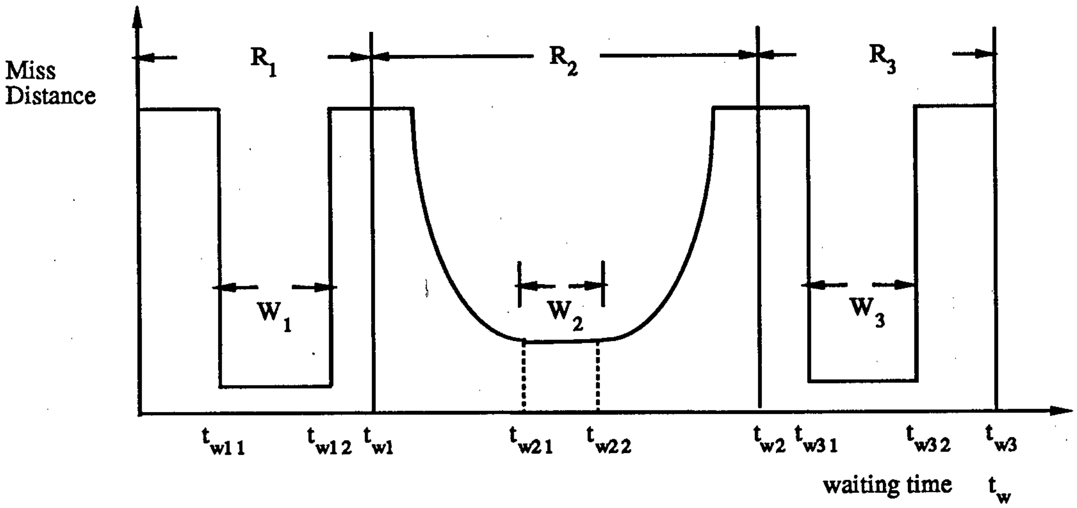


Figure 3.4. Algorithm MD produces three minimal-miss-distance windows, one for each region.

3.1. The Extremal Region R_0 (Module A)

A flowchart for Region R_0 (Module A) is shown in Figure 3.5.

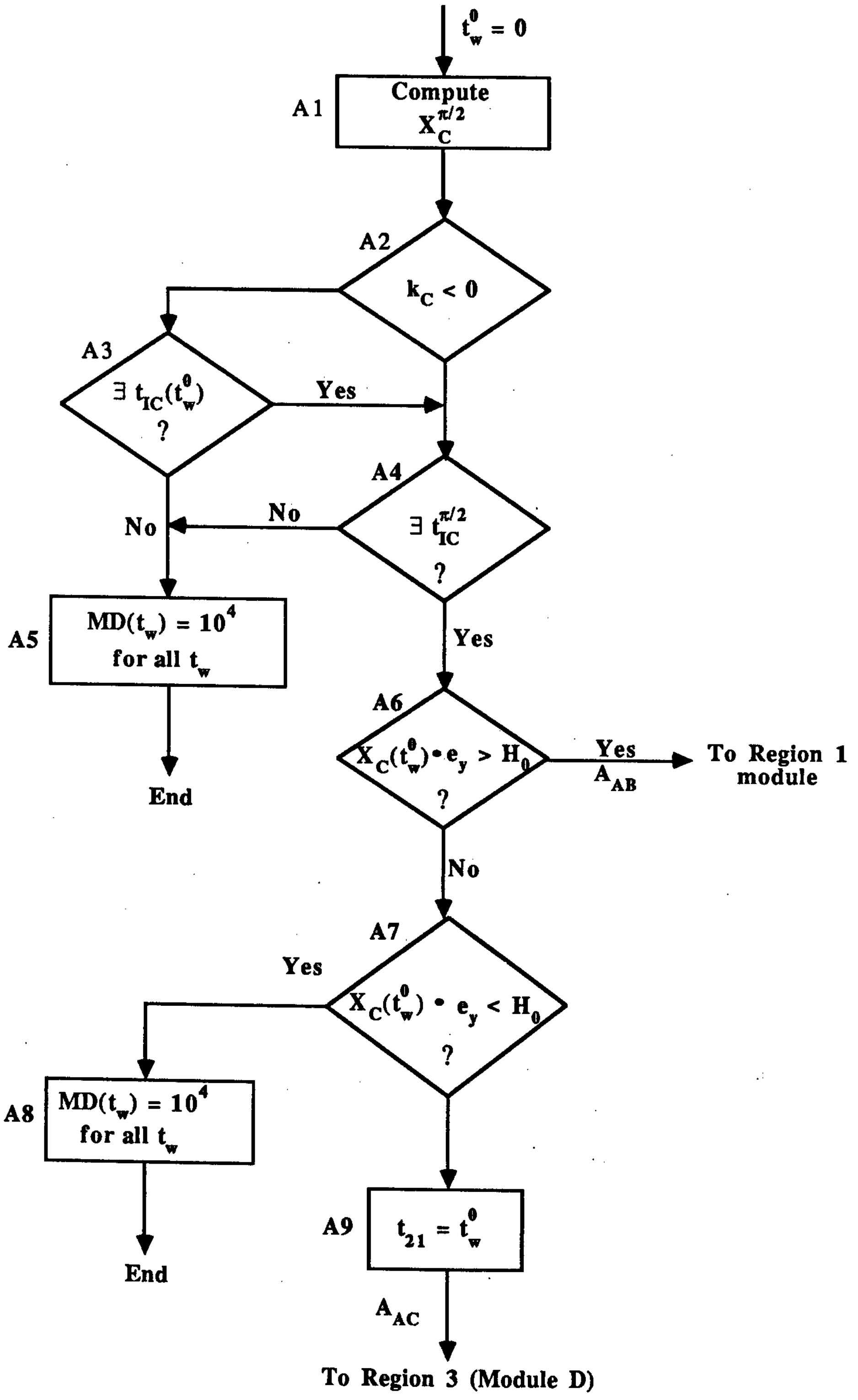

$t_w^0 = 0$
A1
Compute $X_C^{\pi/2}$
A2
$k_C < 0$
A3
$\exists\ t_{IC}(t_w^0)$?
Yes
A4
No
No
$\exists\ t_{IC}^{\pi/2}$?
A5
$MD(t_w) = 10^4$ for all t_w
Yes
End
A6
$X_C(t_w^0) \bullet e_y > H_0$?
Yes
A_{AB}
To Region 1 module
No
A7
Yes
$X_C(t_w^0) \bullet e_y < H_0$?
A8
$MD(t_w) = 10^4$ for all t_w
End
A9
$t_{21} = t_w^0$
A_{AC}
To Region 3 (Module D)

Figure 3.5. Region R_0 is designed to eliminate cases where the MaRV is not reachable for any waiting time t_w (Module A).

Starting with zero waiting time ($t_w^0 = 0$), the point $X_c^{\pi/2}$ on the MaRV trajectory line closest to the interceptor (I) launch location X_I^0 is computed first (A1). If the MaRV (M) is moving *away* from $X_c^{\pi/2}$ (*i.e.*, $k_c = e_{V_M^0} \cdot (X_I^0 - X_M^0) < 0$), *and* if I cannot reach M if launched without delay ($t_w = t_w^0$), then I cannot reach M for any t_w (A2, A3, A5). If M is moving towards the nearest point $X_c^{\pi/2}(k_c \geq 0)$, but $X_c^{\pi/2}$ is not reachable by I, then I again cannot reach M for any t_w (A2 and A4).

If none of the extremal conditions hold, and the intercept point $X_c(t_w^0)$ for t_w^0 is above H_0 (A6), then further analysis is initiated for the exoatmospheric Region (R_1) module *via* control line A_{AB}. If not, keep-out zone penetration is checked in Block A7. If $X_c(t_w^0)$ falls in the keep-out zone (height $X_c(t_w^0) \cdot e_y < H_{ko}$), then the interceptor is ineffective for all t_w (A8). If not, there is no Region R_0, and the lower limit t_{21} of Region 2 (R_2) is set to t_w^0 in Block A9 (see our later discussion of this region) and control is passed to the R_3 module (Module D) through line A_{AD}.

3.2. The Exoatmospheric or Nonmaneuver Region R_1 (Module B)

The purpose of this region is to determine whether there exists an opportunity window where the miss distance is near zero at exoatmospheric altitudes or in regions where the MaRV cannot maneuver (above H_0). Referring to Figure 3.6, the first operation in this module is to determine whether M is reachable by I if I is launched without delay ($t_w = t_w^0$, Block B1). If it is, $MD(t_w^0) = 0$, t_{11} is set to zero ($t_w^0 = 0$), and flag 1 is set since t_{11} is the first value of t_w for which window W_3 opens in Region 3 (B3, B4). If M is not reachable (B1), the miss distance is arbitrarily large (B2) and the clock is incremented by the fixed value of τ seconds (B5).

If the current value of t_w is larger than the maximum time t_F allocated to Region 1 (typically 100 seconds), the MaRV has passed overhead and the algorithm stops, since M never reached below H_0 (B16,B17). Hence, there are no Regions R_2 or R_3, *a fortiori* no windows W_2 or W_3. If t_w does not exceed t_F, and there exists an intercept time $t_{IC}(t_w)$ above H_0, $MD(t_w)$ is set to zero, as discussed in the text (B16, B18, B10, B9). If flag 1 had previously been set, the clock time t_w is again incremented by τ (B8, B5). If not, Window 1 opens at $t_w = t_{11}$, and F_1 is set (B7, B6).

If there does not exist an interception at t_w (B10), $MD(t_w)$ is set to 10^4 (B11). If the window W_1 was previously closed (F_2 set), the waiting clock t_w is incremented by τ (B12, B5). If not, but W_1 was previously opened (F_1 set), we have reached the end of W_1, t_{12} is set to t_w (B13, B14), and W_1 is now closed (B15).

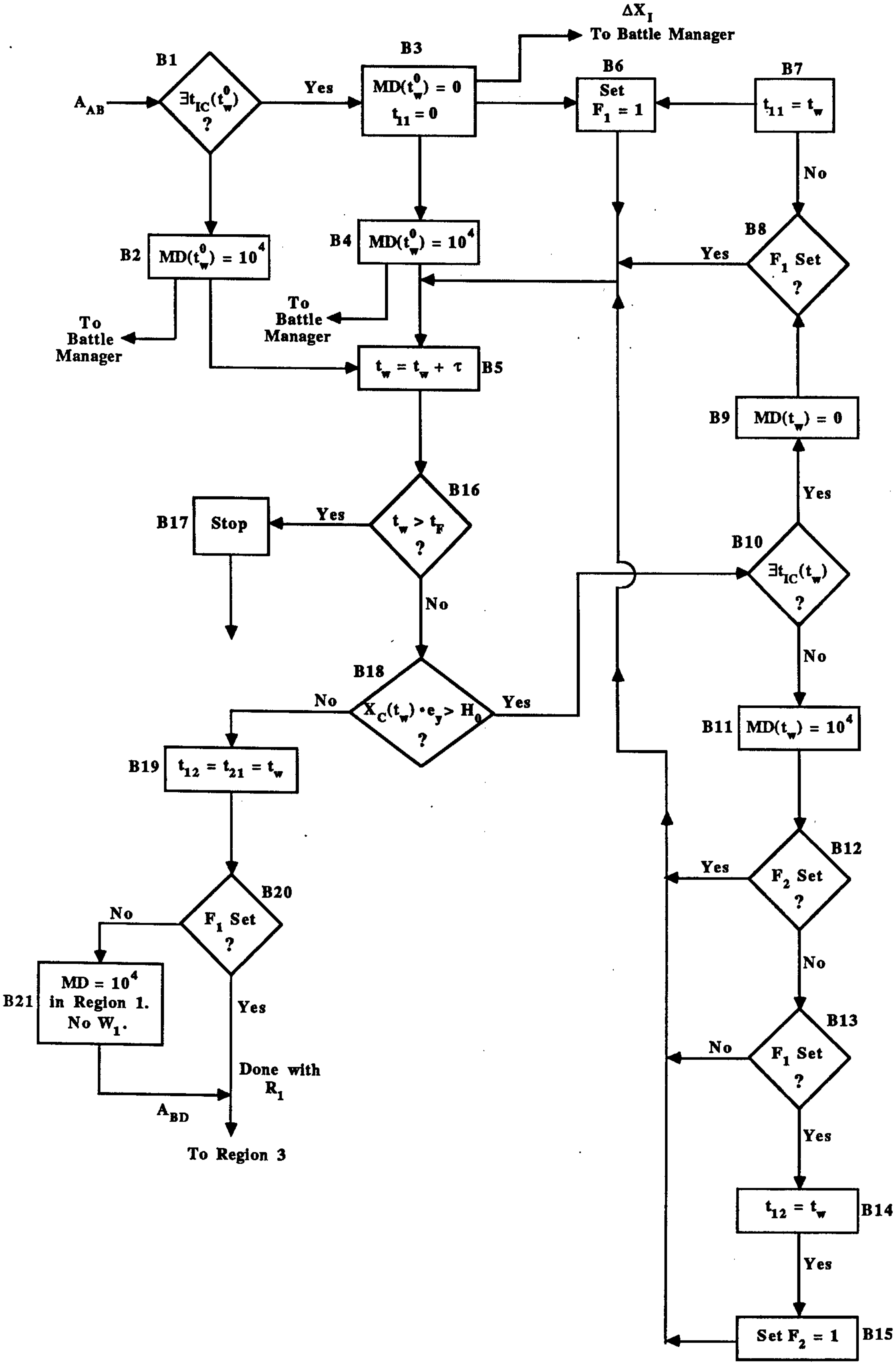


Figure 3.6. Flowchart for the exoatmospheric region R_1 (Module B).

When the MaRV falls below H_0 (B18) and window W_1 was never opened (B19), there is no window in Region 1 (B20). If it is still open, we have reached the end of R_1, hence the end of W_1, and the beginning t_{21} of window W_2 (B21). We then analyze region R_3 by passing control to Module D *via* control line A_{BD}.

3.3. The Tailchase Region R_3 (Module D)

Recall that we have a tailchase condition whenever $V_M^0 \cdot (X_c(t_w) - X_I^0) > 0$, *i.e.*, any time the interceptor velocity has a strictly positive component along the MaRV velocity.

Module D receives two inputs, one (A_{AD}) from Module A and another (A_{BD}) from Module B, as shown in the flowchart of Figure 3.7. It starts by resetting the flags previously used in other modules (D1). If the MaRV has reached the keep-out zone (D2) prior to being engaged in a tailchase (D2), then the current value of t_w equals the closing time (t_{22}) of window W_2 (D3), there is no Region 3 (D4), and the waiting clock is set to the starting time t_{21} for W_2 in Region 2. If there is no W_2, we are done (D6, D7). If not, we explore R_2 with Module 3 (D26).

If M is still outside the keep-out zone (D2), but we do not yet have a tailchase (D8), we need to explore future conditions. To accomplish this efficiently, an accelerated waiting clock is used in which the increment value is 10τ, and not τ as before (D9). When the first tailchase time is reached on that "rough" clock (D8), the waiting time t_w is reset to $t_w - 10\tau$, and the start t_{22} of Region 2 is set to t_w (D10, D11). Using this rough clock is obviously ten times faster than using the finer clock increment τ.

If the speed of the interceptor exceeds that of the MaRV in the tailchase scenario, $MD(t_w) = 0$ (D12, D13). If the window W_3 was not yet opened (D14), it is now opened (D16) and its opening time t_{31} equals t_w (D15). If W_3 was previously opened, the clock is incremented by τ (D17).

If M is now faster than I (D12) and if W_3 is currently open (D18), it is now closed (D19) and control is passed to Module 3 to explore Region R_2 after resetting F_1 (D20, D6, D21).

If, after incrementing the clock (D17), the MaRV is still above the keep-out altitude H_{ko} (D22), a new relative velocity check is made (D12) and that loop is re-entered. If M has fallen below H_{ko}, and if F_1 was never set, there is no W_3, and control is transferred to Module 3 (D22, D23, D25, D20, D21, D6, D26). If F_1 had been set, the amount of waiting time t_w equals the closing time t_{32} of W_3 (D24) and Module 3 takes over, as before.

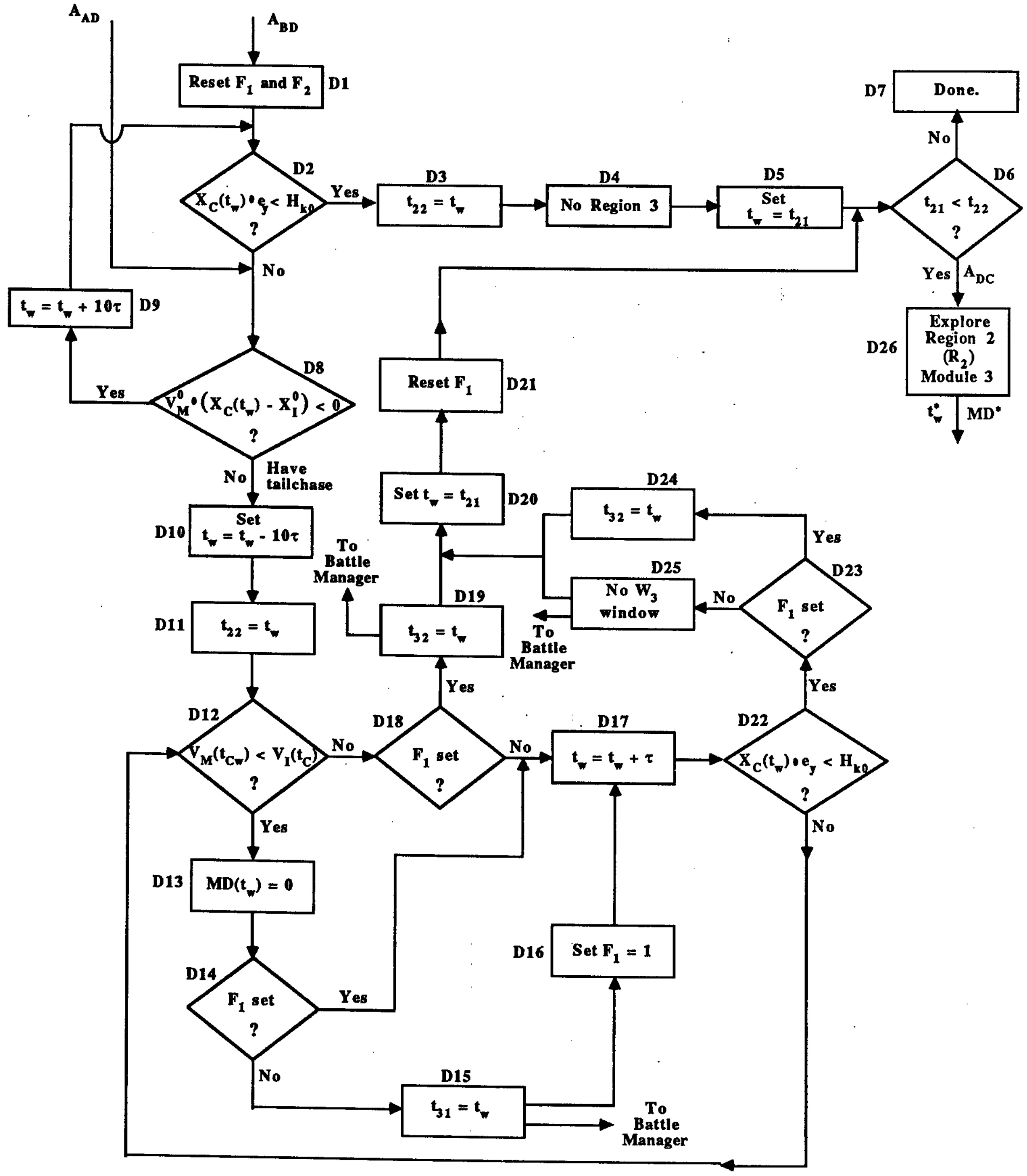


Figure 3.7. Flowchart for the tailchase region R_3 (Module D).

3.4. The Intermediate Region R_2 (Module C)

In this section, we discuss the most complex engagement region, region R_2. Because the algorithms dedicated to this region are based in part on heuristic arguments which are not obvious, we have devoted a special introductory section (Section 3.4.1) to presenting and defending these arguments. Then, in Section 3.4.2, we develop the mathematical machinery required to define miss distance in Region 2. Finally, in Section 3.4.3 we present and discuss the flowchart for Module C, as we did in previous sections for the other regions.

3.4.1. INTRODUCTION

Two principal characteristics distinguish the intermediate region (R_2) from the other regions. First, engagements in this region take place endoatmospherically (below H_0), where the atmosphere is sufficiently dense that drag and lift become significant, and evasive maneuvering is the principal method for avoiding interception. Secondly, and by definition, tailchasing does not arise in this region, and the closure rate between MaRV and interceptor is thus positive.

Since maneuvering capabilities are such strong determinants in this region, and little is known about the future maneuvering tactics of a MaRV, how then can miss distances be accurately predicted in this region? This question has obviously long been of significant interest to many analysts and interceptor designers, and several analyses and methods [13–19] have been reported.

In the context of battle management, one popular method, summarized in [19], is based on a combinatorial approach where, on the basis of intelligence data and various technological constraints, representative MaRV maneuvers and trajectories are derived. Interceptor capabilities against such trajectories are then evaluated using computer simulation methods. In spite of its popularity, disadvantages of this method appear obvious, particularly for battle management purposes. First, there is an uncountable infinity of trajectories, and to evaluate, design, or deploy an interceptor on the basis of just a few of these is very risky. Even if these few contain some "optimal" trajectory, this approach encourages gaming by the adversary, and may result in his selecting a trajectory which is suboptimal by the assumed criterion, but optimal by another—equally rational—criterion. Second, this method does not sufficiently account for real-time information that is available from sensors as the engagement evolves. Reliable state information about a MaRV's progress towards its target for instance will significantly reduce its maneuvering choices, particularly as it nears the target. Third, recall that the principal reason why the battle manager needs miss distance information is to effectively allocate interceptor type and quantity, launch time, and launch direction. Traditional miss distance prediction methods are of little assistance in making such decisions.

Another approach (see Gavel [20], for instance) employs optimal control methods to obtain some "optimal" trajectory which can be used in simulations to derive miss distance estimates. While this method is still under investigation, it appears to have limitations relating to the choice of a criterion and to the uniqueness of a solution, given that engagement dynamics are highly nonlinear in this region. Furthermore, expressing the problem as a multidimensional optimization problem has severe real-time computational limitations, particularly for engagement scenarios where the MaRV does not in fact follow a precomputed optimal trajectory, and real-time adaptation must be considered.

Another method is popular in the guidance community, one which uses linear dynamical models of both vehicles to estimate miss distances, as exemplified by Reference [14].

In contrast to these methods, our approach is based on an evasion-and-capture test whose outcome is determined by the maximal technological capabilities attributed to both vehicles. Whereas other methods rely on detailed tactical assumptions and on specific guidance algorithms to estimate miss distances, and could thus be classified as "scenario-based," we have intentionally avoided such assumptions in developing our test, and our approach should probably be classified as "technology-based." Its basic premise is rather simple, and can be characterized by a single question:

> Assuming a "reasonably rational and informed MaRV-Interceptor pair," what is the "best" that the interceptor can do to capture the MaRV, given the current state and environment of both?

To see what we mean by "reasonably rational and informed" and by "best," refer to Figure 3.8, where we show a typical scenario in which a MaRV attempts to evade an interceptor approaching at an angle θ_c with a single right-hand turn, which the interceptor counters with a corresponding left-hand turn. We say that the MaRV-Interceptor pair is *reasonably informed* if they both possess some state information about each other throughout the engagement. In the case of the MaRV, this assumption deserves some discussion because the MaRV is conventionally assumed to operate in an open-loop mode, unaware of the interceptor. As we shall see, this is not important in our minimax launch logic because an interceptor is launched at the precise time at which the maximum miss distance is a minimum. The location of this minimum is determined by atmospheric and dynamic conditions unrelated to the information state of the MaRV.

Although the *location* of the minimum miss distance on the launch time scale is insensitive to the MaRV information state, its *magnitude* generally is not. This may lead to considerably overestimating the *size* of the minimum miss distance, particularly if the MaRV is *known* not to maneuver at all. Such an error would lead the interceptor manager to underestimate the probability of kill, and might cause an excessive expenditure of interceptors in special cases so favorable to the defense. But such MaRV disabilities are virtually impossible to predict in practice, hence have little operational value from a defense allocation point of view. Whether the MaRV has interceptor state information or whether it follows a preprogrammed trajectory, the defense cannot predict its maneuver, and must assume that it will maneuver to maximize its distance from the interceptor. It is this maximum distance which is minimized in the MD algorithm.

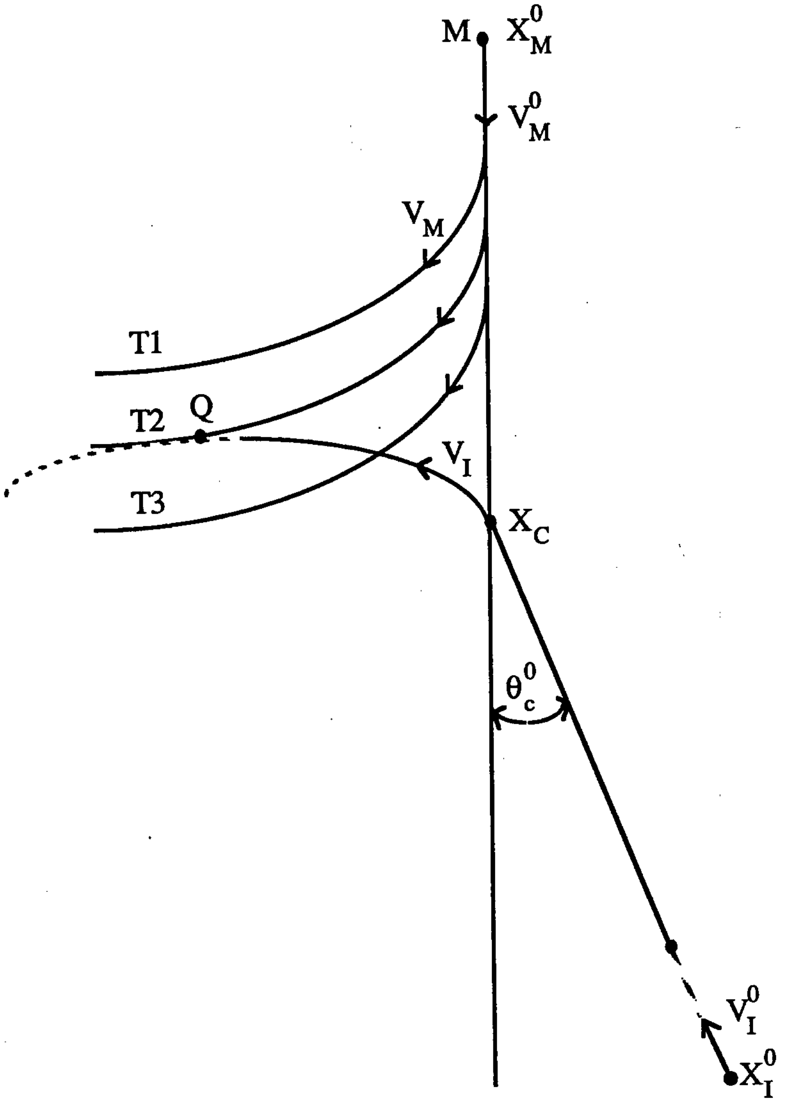


Figure 3.8. Illustrating a single-turn evasive maneuver.

Delays in interceptor response can be major contributors to miss distance. Not only is MaRV state information delayed when it arrives at the interceptor, but the guidance and control subsystem of the interceptor cannot respond instantaneously. In Section 3.4.2 below, we discuss how the MD algorithm accounts for the causes and effects of such delays. State information received by the interceptor can also be severely distorted by various error mechanisms in the propagation and processing of information. To account for the effects of random errors and information distortion, we develop a sensitivity and error propagation model in Section 3.4.

We say that a MaRV and an interceptor are *reasonably rational* if neither maneuvers too early or too late, based on their knowledge of each other's state. Referring to Figure 3.8, a MaRV following trajectory $T1$ has turned too early, since it gives the interceptor an opportunity to "goal-tend," a maneuver providing significant leverage to the interceptor. If M delays its evasive maneuver too much, and takes trajectory $T3$, another advantage is given the interceptor since the latter need not turn as much or as quickly to intercept M. A dramatic—and admittedly extreme—example of this advantage is where M delays its turn so long that it would forfeit any opportunity to maneuver before intercept, which would take place at X_c. Normally, this would arise only if M behaved ballistically, as if it were an ordinary RV. From a trajectory point of view, a similar advantage is given the MaRV if the interceptor turns too late. The MaRV then escapes along trajectory $T3$.

So the only scenario which is equitable to both the MaRV and the interceptor is the second trajectory ($T2$), which touches the interceptor trajectory at one point Q. The location of both vehicles along their respective locus at any given point in time obviously depends on their velocity and the system time delays experienced by the interceptor. Using the above assumptions, the miss distance is thus determined only by the system time delay and by the relative velocity and acceleration capabilities of the vehicles, and not by tactics or strategy, and that is why we describe our approach as *technology-based.*

An equally compelling reason why both the MaRV locus and the interceptor locus must meet at only one point follows from the requirements for mathematical consistency. Clearly, if both vehicles have exactly the same capability and errors, and delays are neglected, the miss distance should be zero since we do not allow tactical advantages. Futhermore, if the velocity or acceleration capabilities of the interceptor are increased, the miss distance could not possibly get worse, and should remain at zero. But, if the two trajectories (loci) do not remain in contact as interceptor capabilities are increased, this is impossible. Since the late-maneuver ($T3$) was disallowed earlier, the two loci must meet at one and only one point.

Having developed and defended our fairness doctrine for a single-turn, we now proceed to show that the outcome of an entire engagement—and hence the miss distance between the MaRV and the interceptor—may be predicted from this simplified engagement.

Regardless of its complexity, a trajectory is a sequence of turns, potentially infinite. It is well known, however, that considerable energy and momentum is lost during a maneuver, and that long and complex maneuvers are not effective. What current technology encourages is a trajectory consisting of one or two carefully timed turns, the second of which typically takes place at reduced velocities and altitudes where the interceptor usually has a speed or maneuvering advantage. Considering that the battle manager employs a two-layer shoot-look-shoot strategy where, if a MaRV is missed during a first engagement with an overlay interceptor, an underlay interceptor is launched to engage it during its second maneuver, this maneuvering advantage further reduces our concern for the MaRV's second maneuver. The relative capabilities of both vehicles can thus be estimated using very simple maneuvers, sometimes a single turn [16].

Recall, however, that any engagement outcome is strongly influenced by where—and thus when—the engagement takes place. A single test at a fixed altitude would not properly represent all reasonable possibilities, and would be comparable to the combinatorial methods which we criticized earlier. Instead, our algorithm sweeps *all* altitudes *via* the waiting variable t_w, to determine when the predicted miss distance is a minimum. At this minimum, called t_w^*, the interceptor is said to do *best*. We thus literally examine an infinite set of trajectories and engagements before settling on a miss distance. We claim that this infinite set adequately represents *all* reasonable MaRV maneuvers.

3.4.2. MATHEMATICAL ANALYSIS OF REGION 2

Now we develop the additional machinery required to derive miss distance predictions for Region R_2. Referring to Figure 3.9, our objective is to find an expression for the minimum distance between the MaRV and the interceptor, as both are negotiating maximal turns. Roughly speaking, our algorithm involves three major steps. In the first step, it computes the interception point X_c based on the continuation of the MaRV velocity V_M^0. The derivation of X_c was discussed in detail in Section 2.4.2. The second step is to construct the MaRV and interceptor loci. Although we show the two loci touching each other in Figure 3.9, this iterative construction process starts with a separation Δ, as illustrated in Figures 2.6 and 3.10, and is designed to reduce Δ to near zero in a minimal number of steps. Because of the severe atmospheric and dynamical nonlinearities in our vehicle model, transcendental equations must be solved. Since such equations have no satisfactory closed-form approximations, we are forced to employ numerical optimization methods to solve for the unique point where both trajectories meet, when their separation Δ has been reduced to zero. Given that the starting point in any nonlinear optimization problem is crucial to the accuracy and efficient convergence of the optimization, a considerable amount of effort was devoted to choosing this initial condition. By neglecting atmospheric nonlinearities for this initial condition, we were able to exploit the fact that both vehicles travel along the perimeter of a circle, and obtained a closed-form solution which is exact when atmospheric effects are neglected.

More formally, consider Figure 3.11, where all the initialization parameters are illustrated. For this approximation, the loci of both vehicles are circles which are assumed to touch only at one point. The distance $d^0_{MI}(t_w, \Delta t^0_I, \theta^0_c)$ between M and I at the time

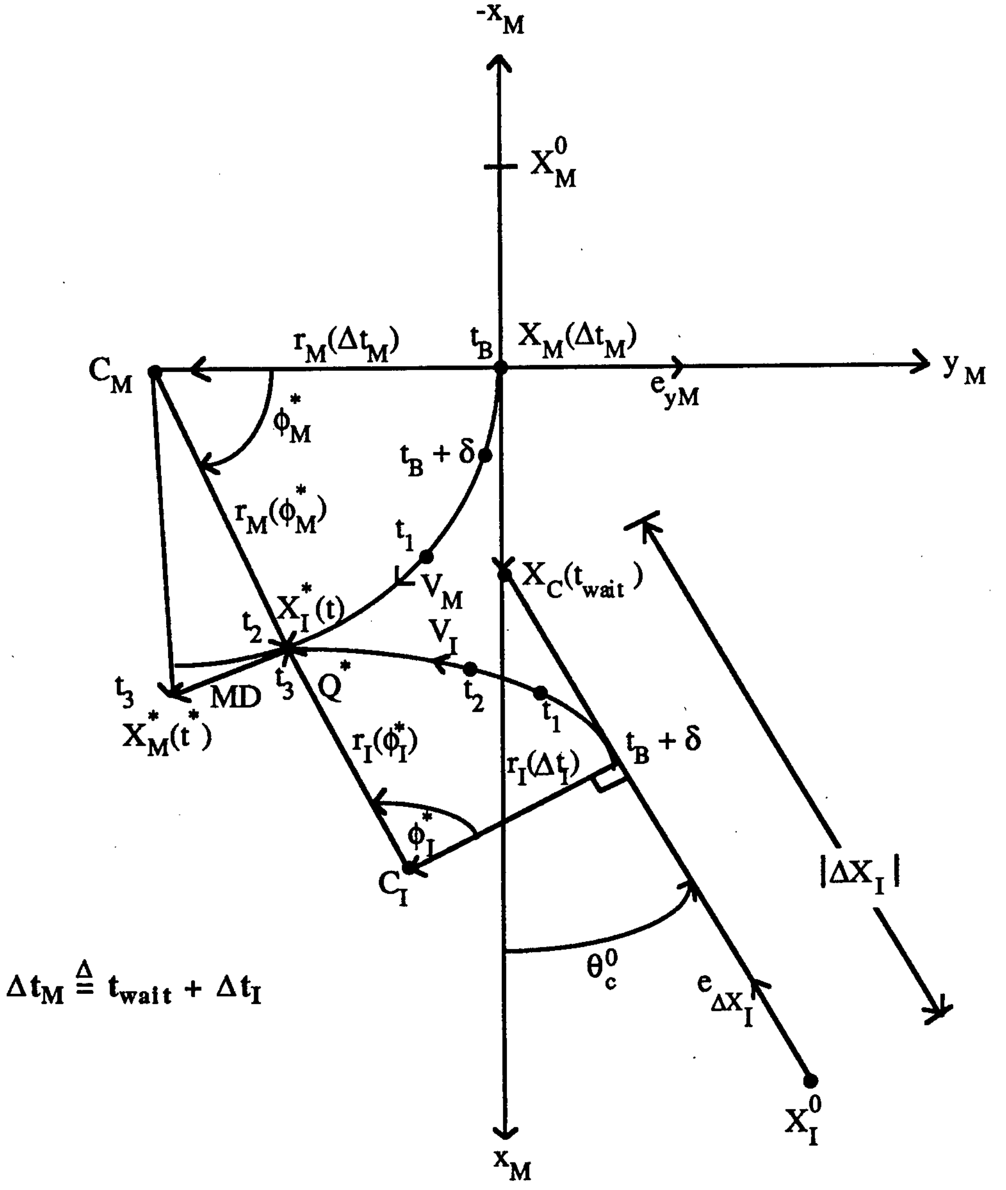


Figure 3.9. Miss distance geometry.

$\Delta t_M^0 = \Delta t_I^0 + t_w$ when both are initiating their maneuvers ($\delta = 0$ for this initialization) and when the collision angle is θ_c^0, is then

$$d_{MI}(t_w,\, \Delta t_I^0,\, \theta_c^0) = 2\left[|r_M(\Delta t_M^0)||r_I(\Delta t_I^0)|\right]^{1/2} \cos\theta_c^0 + r_I(\Delta t_M^0)\sin(\theta_c^0) \quad . \qquad (3.2)$$

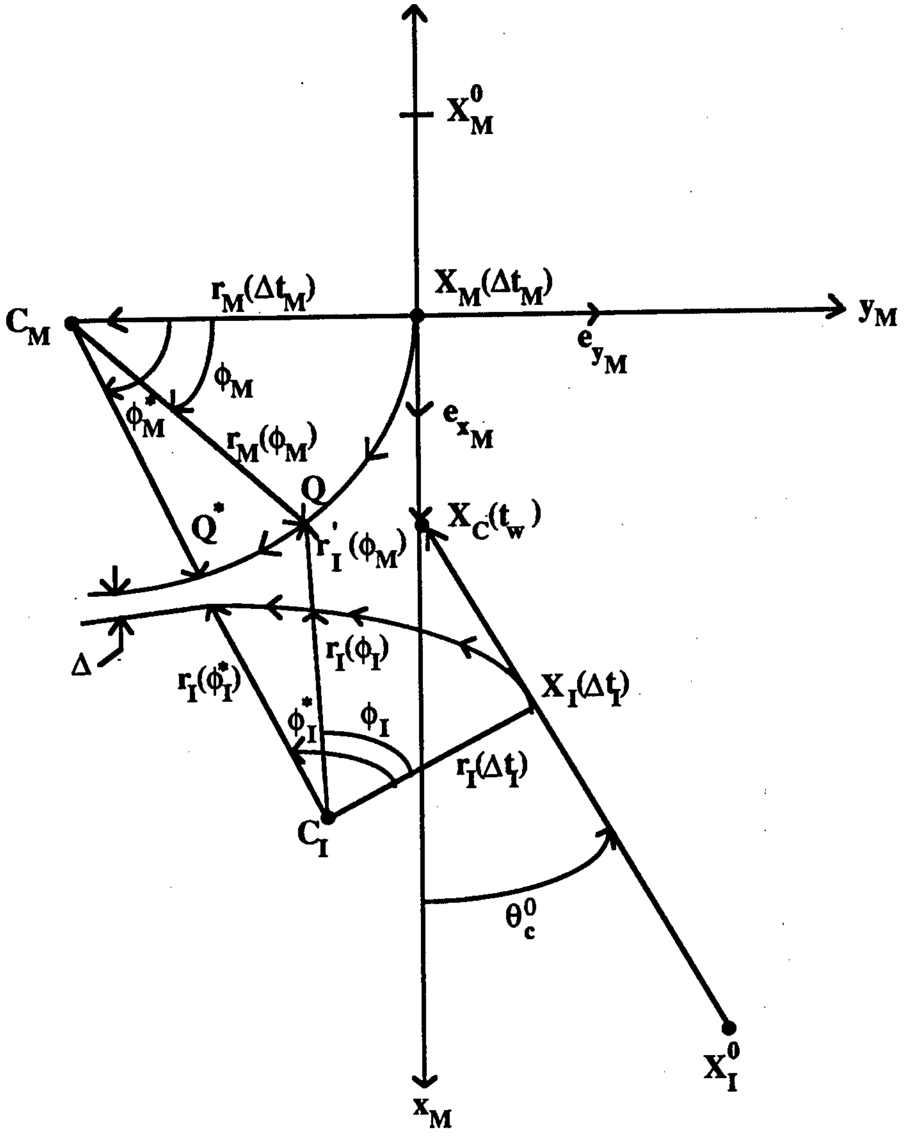


Figure 3.10. Geometric relationship between optimization variables showing the initial offset Δ.

Obtained by purely geometric considerations, this distance must equal the distance $R^0_{MI} - \Delta R_{MI}$ separating the two vehicles after M has traveled for $\Delta t^0_M = t_w + \Delta t^0_I$ seconds and I has traveled for Δt^0_I seconds, where R^0_{MI} is defined in Figure 3.11, and

$$\Delta R_{MI}(t_w, \Delta t^0_I) = \int_0^{\Delta t_I{}^0 + t_w} |V_M(\tau)| d\tau + \int_0^{\Delta t_I{}^0} |V_I(\tau)| d\tau \quad . \tag{3.3}$$

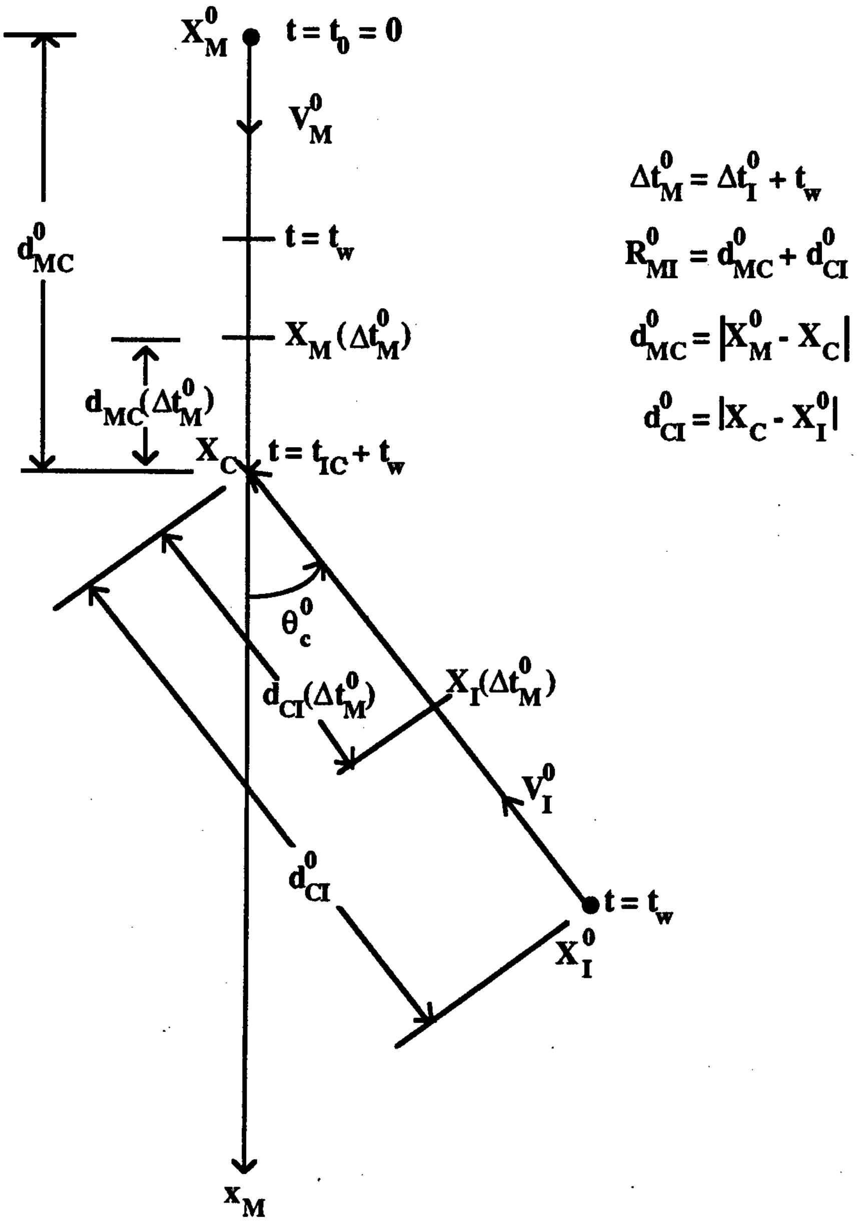


Figure 3.11. Parameters and geometry for the initialization process.

But Δt_I^0 is not known, hence must be solved for.. So the initialization of the control variable Δt_I may be reduced to solving the equation

$$F_0(t_w,\, \Delta t_I^0,\, \theta_c^0) = d_{MI}(t_w,\, \Delta t_I,\, \theta_c^0) - R_{MI}^0(t_w) + \Delta R_{MI}(t_w,\, \Delta t_I^0) = 0 \tag{3.4}$$

for Δt_I. The solution to this equation is found by using the root solving procedure RTSAFE ([10] and Appendix F) and is defined as Δt_I^0.

Starting with this initial estimate, all that remains is to adjust both loci to account for atmospheric effects, and this is done with Algorithm DBRENT ([10] and Appendix E), and with a modified form of the Newton-Raphson approach. Realizing that the principal independent variable to slide the two loci towards or away from each other is the time-to-go Δt_I (assuming the waiting time t_w is fixed), the algorithm computes the partial derivative $\partial\Delta(t_w)/\partial\Delta t_I$, where $\Delta(t_w)$ is the distance between both loci. Then, after computing $\Delta(t_w)$ using DBRENT, a correction value $d(\Delta t_I(t_w)) = \Delta(t_w)[\partial\Delta(t_w)/\partial\Delta t_I]^{-1}$ for $\Delta t_I(t_w)$ is used to adjust the time-to-go accordingly, thereby sliding the loci to reduce Δ.

This iterative process continues until Δ falls below an acceptance threshold ϵ, which was arbitrarily set to 10 feet. Because all the required derivative information is available in closed analytical form (see Appendix E), this process converges very rapidly; in a *single step* for the vast majority of cases tested. In some rare instances, several steps are needed, and we have set a limit of 5 on the number of steps allowed. The penalty resulting from this limit has never exceeded 10.0 feet of error in our miss distance tests.

The two loci now joined, the third and final step in the procedure is to compute the miss distance. Referring again to Figure 3.9, the MaRV initiates its turn at the "banking time" t_B, and the interceptor at a delayed time $t_B + \delta$, where δ represents the various interceptor time delays. In this example the MaRV reaches Q^* ahead of the interceptor, at time $t_2 = t_M^*$. The interceptor reaches Q^* at time $t_3 = t_I^*$, when the MaRV is ahead at position $X_M^*(t_I^*)$. The miss distance estimate is now defined as

$$MD(t_w) = \begin{cases} \min_{t_B \le t \le t_{M^*}}\{|X_M(t+\delta) - X_I(t)|\}\,, & t_M^* < t_I^* + \delta \\ 0\,, & t_M^* \ge t_I^* + \delta \end{cases}\,, \tag{3.5}$$

where:

t_B is the time at which the MaRV initiates its turn,

t_M^* is the time taken by M to reach Q^*,

t_I^* is the time taken by I to reach Q^*,

δ is the interceptor system time delay.

To provide a helpful comment, the miss distance should be zero if the interceptor reaches Q^* before the MaRV does since the interceptor can then turn towards the MaRV and intercept it before it reaches Q^*. If the MaRV gets to Q^* first, the validity of the first expression is rather obvious, since $MD(t_w)$ can only get larger beyond Q^*. As a good approximation to $MD(t_w)$ in this condition, and to save computation time, Algorithm MD actually computes the simpler expression

$$MD(t_w) = |Q^* - X_M(t_I^*)| \quad , \tag{3.6}$$

where $X_M(t_I^*)$ is the position of M at the precise time t_I^* when I reaches Q^*.

For a more rigorous derivation of the distance $\Delta(t_w)$ between both trajectories, refer again to Figure 3.10, and consider some arbitrary point Q on M's trajectory or locus. Then

$$\Delta(t_w) = \min_{\phi_M} \left\{ \left[|r_I'(\phi_I(\phi_M))| - |r_I(\phi_I(\phi_M))| \right] \right\} \quad , \tag{3.7}$$

where $\phi_I(\phi_M)$ is the angle swept by I as it travels from the start of its turn at $X_I(\Delta t_I)$ to the point where its extended radius of curvature passes through Q, as shown. Clearly $\phi_I(\phi_M)$ is indeed determined by ϕ_M only. We define as ϕ_M^* the MaRV angle ϕ_M at which this minimum occurs. Observe also that this minimum could be negative, in which case the loci must be pulled away from each other.

Using now the MaRV coordinate system $M = (x_M, y_M, z_M)$, and standard vector analysis methods,

$$Q = X_M(\Delta t_M) - |r_M(\Delta t_M)| e_{y_M} + r_M(\phi_M) \tag{3.8}$$

and

$$Q = X_M(\Delta t_M) + d_{MC}(t_w, \Delta t_I) e_{x_M} - d_{CI}(t_w, \Delta t_I) e_{\Delta X_I} + r_I(\Delta t_I) e_{z_M} \times e_{\Delta X_I} + r_I'(\phi_M) \quad , \tag{3.9}$$

where:

Δt_I	is the interceptor flight time,
t_w	is the launch waiting time,
$\Delta t_M \triangleq t_w + \Delta t_I$	is the MaRV flight time,
ΔX_I	is the interceptor launch direction vector,
ϕ_M	is the angle swept by M as it travels from $X_M(\Delta t_M)$ to Q.

But

$$r_M(\phi_M) = |r_M(\phi_M)| (\cos\phi_M) e_{y_M} + |r_M(\phi_M)| (\sin\phi_M) e_{x_M} \quad , \tag{3.10}$$

$$e_{\Delta X_I} = -(\cos\theta_c^0) e_{x_M} - (\sin\theta_c^0) e_{y_M} \triangleq e_{V_I^0} \quad , \tag{3.11}$$

and

$$|r_I(\Delta t_I)|e_{z_M} \times e_{V_I^0} = |r_I(\Delta t_I)|\,(\sin\theta_c^0)e_{x_M} - |r_I(\Delta t_I)|\,(\cos\theta_c^0)e_{y_M} \quad , \tag{3.12}$$

where:

θ_c^0 is the incidence (collision) angle,

$r_M(\phi_M)$ is the curvature radius of M as a function of its angular displacement ϕ_M,

$r_I(\Delta t_I)$ is the curvature radius of I as a function of its flight time Δt_I,

$r_M(\Delta t_M)$ is the curvature radius of M as a function of its flight time Δt_M.

Since

$$e_{z_M} \times e_{V_I^0} = (\sin\theta_c^0)e_{x_M} - (\cos\theta_c^0)e_{y_M} \quad , \tag{3.13}$$

$$d_{MC}(t_w, \Delta t_I) = |X_M(\Delta t_M) - X_c(t_w)| = |X_M^0 - X_c(t_w)| - \int_0^{\Delta t_M} |V_M(\tau)|\,d\tau \tag{3.14}$$

and

$$d_{CI}(t_w, \Delta t_I) = |X_c(t_w) - X_I(\Delta t_I)| = |X_I^0 - X_c(t_w)| - \int_0^{\Delta t_M} |V_I(\tau)|\,d\tau \quad . \tag{3.15}$$

Hence

$$\begin{aligned} Q &= X_M(\Delta t_M) - |r_M(\Delta t_M)|e_{y_M} + |r_M(\phi_M)|\,(\cos\phi_M)e_{y_M} + |r_M(\phi_M)|\,(\sin\phi_M)e_{x_M} \\ &= X_M(\Delta t_M) + |r_M(\phi_M)|\,(\sin\phi_M)e_{x_M} \\ &\quad + [|r_M(\phi_M)|\,\cos(\phi_M) - |r_M(\Delta t_M)|]e_{y_M} \quad . \end{aligned} \tag{3.16}$$

Therefore:

$$\begin{aligned} r_I'(\phi_M) = &-|r_M(\Delta t_M)|e_{y_M} + |r_M(\phi_M)|\,(\cos\phi_M)e_{y_M} + |r_M(\phi_M)|\,(\sin\phi_M)e_{x_M} \\ &- d_{MC}(t_{\text{wait}}, \Delta t_I)e_{x_M} - d_{CI}(t_{\text{wait}}, \Delta t_I)\,(\cos\theta_c^0)e_{x_M} + |r_I(\Delta t_I)|\,(\cos\theta_c^0)e_{y_M} \\ &- d_{CI}(t_{\text{wait}}, \Delta t_I)\,(\sin\theta_c^0)e_{y_M} - |r_I(\Delta t_I)|\,(\sin\theta_c^0)e_{x_M} \\ &= Ae_{x_M} + Be_{y_M} \quad , \end{aligned} \tag{3.17}$$

where

$$\begin{aligned} A = |r_M(\phi_M)|\,(\sin\phi_M) - d_{MC}(t_{\text{wait}}, \Delta t_I) - d_{CI}(t_{\text{wait}}, \Delta t_I)\,(\cos\theta_c^0) \\ - |r_I(\Delta t_I)|\,(\sin\theta_c^0) \quad , \end{aligned} \tag{3.18}$$

and

$$B = |r_M(\phi_M)|\,(\cos\phi_M) - |r_M(\Delta t_M)| - d_{CI}(t_{\text{wait}},\, \Delta t_I)\,(\sin\,\theta_c^0)$$
$$+\,|r_I(\Delta t_I)|\,(\cos\theta_c^0) \quad . \tag{3.19}$$

To obtain $r_I(\phi_M)$, it now suffices to compute $\phi_I(\phi_M)$ and $r_I\,(\phi_I(\phi_M))$.

To find ϕ_I,

$$\begin{aligned}\cos(\phi_I) &= \frac{-r_I'\bullet(e_{z_M} \times e_{\Delta X_I})}{|r_I'|} = \frac{-r_I'\bullet e_{z_M} \times e_{V_I^0}}{|r_I'|} \\ &= \frac{r_I'(e_{x_M},\, e_{y_M},\, e_{z_M})\bullet\,\left(e_{z_M} \times \left(\cos\theta_c^0\right) e_{x_M} + (\sin\theta_c^0)e_{y_M}\right)}{|r_I'|} \\ &= \frac{r_I'(e_{x_M},\, e_{y_M},\, e_{z_M})\bullet\,\left(\cos\theta_c^0\right) e_{y_M} - \left(\sin\theta_c^0\right) e_{x_M}}{|r_I'|} \\ &= \frac{(Ae_{x_M} + Be_{y_M})\bullet\,\left(\cos\theta_c^0\right) e_{y_M} - \left(\sin\theta_c^0\right) e_{x_M}}{|r_I'|} \\ &= \frac{-A\,(\sin\theta_c^0) + B\,(\cos\theta_c^0)}{|r_I'|} \quad .\end{aligned} \tag{3.20}$$

Hence,

$$\cos(\phi_I) = \frac{-A\,(\sin\theta_c^0) + B\,(\cos\theta_c^0)}{|r_I'|} \quad . \tag{3.21}$$

To find $r_I\,(\phi_I\,(\phi_M))$, we now simply use Equations (2.27) and (2.28).

To compute the miss distance, we need the time t_I^* at which the interceptor reaches Q^*. But the angle ϕ_I^* swept by I to reach Q^* was found by numerical methods earlier (Equations (3.2)–(3.4), and with Equation (3.21). We can thus find t_I^* by solving

$$\phi_I^* = \int_{t_B}^{t_I^*} \frac{V_I(\tau)}{r_I(\tau)}\, d\tau \quad . \tag{3.22}$$

To avoid a difficult nonlinear integration, we approximate $r_I(\tau)$ by $\left[r_I(t_B) \times r_I(\phi_I^*)\right]^{1/2}$. This is a very good approximation since r_I does not vary much during a sharp turn, particularly at low altitudes where r_I is small and thus the loss of altitude during a bank will also be small. This function $V_I(\tau)$ and its integral are derived in Appendix B.

Continuing, if the interceptor reaches Q^* before the MaRV does, the miss distance is zero, since it can simply turn towards the MaRV as it approaches Q^*. Mathematically, if

$$\int_{\Delta t_M^0 = \Delta t_I^0 + t_w}^{\Delta t_M^0 + t_{BI}^* + \delta} \frac{V_M(\tau)d\tau}{\left[r_M(\Delta t_M^0) r_M(\phi_M^*)\right]^{1/2}} < \phi_M^* \quad , \qquad \text{then } MD = 0 \quad . \tag{3.23}$$

If not, we need to compute the time t^*_{BM} required by M to reach Q^*, by solving

$$\phi^*_M = \int_{\Delta t^0_M}^{\Delta t^0_M + t^*_{BM}} \frac{V_M(\tau)d\tau}{\left[r_M(\Delta t^0_M) r_M(\phi^*_M)\right]^{1/2}} \tag{3.24}$$

for t^*_{BM} and evaluating

$$MD = \int_{\Delta t^0_M + t^*_{BM}}^{\Delta t^0_M + t^*_{BI} + \delta} V_M(\tau)\, d\tau \quad . \tag{3.25}$$

A closed-form solution for t^*_{BM} can be obtained as follows:

$$\begin{aligned} \phi^*_M(t^*_{BM}) &= \int_{\Delta t^0_M}^{\Delta t^0_M + t^*_{BM}} k_1[1 - k_2\tau]\, d\tau = k_1\tau \Big|_{\Delta t^0_M}^{\Delta t^0_M + t^*_{BM}} - \frac{k_1 k_2 \tau^2}{2} \Big|_{\Delta t^0_M}^{\Delta t^0_M + t^*_{BM}} \\ &= k_1 t^*_{BM} - \frac{k_1 k_2}{2}[t^{*2}_{BM} + 2\Delta t^0_M t^*_{BM}] \quad . \end{aligned} \tag{3.26}$$

In standard quadratic form:

$$\frac{k_1 k_2}{2}\, t^{*2}_{BM} + (\Delta t^0_M k_1 k_2 - k_1) t^*_{BM} + \phi^*_M = 0 \quad .$$

Solution:

$$t^*_{BM} = \frac{(1 - \Delta t^0_M k_2) - \sqrt{(1 - \Delta t^0_M k_2)^2 - 2\frac{k_2}{k_1}\phi^*_M}}{k_2} \quad , \tag{3.27}$$

where

$$\text{if} \qquad \tau_0 = \frac{H_0 - X^0_M \bullet e_y}{V^0_M \bullet e_y} < 0 \quad ,$$

$$k_1 = \frac{|V^0_M|}{[r_M(\Delta t^0_M) r_M(\phi^*_M)]^{1/2}} \quad \text{and} \quad k_2 = \frac{1}{K_{DM}} \quad ,$$

and if $\tau_0 \geq 0$,

$$k_1 = \frac{|V^0_M| \left[1 + \frac{\tau_0}{k_{DM}}\right]}{\left[r_M(\Delta t^0_M) r_M(\phi^*_M)\right]^{1/2}} \quad \text{and} \quad k_2 = \frac{1}{[k_{DM} + \tau_0]} \quad .$$

3.4.3. UNCERTAINTY ANALYSIS AND ERROR PROPAGATION

Although parameter and state vector errors can affect miss distance estimates in all three regions, Regions 1 and 3 are rather insensitive to such errors because of their binary nature. Miss distances tend to be either very high or very low in these regions, and only the width of their windows can be affected. Battle management decisions are rarely sensitive to this width.

In Region 2, however, state vector or parameter errors can be significant, and in this section we develop a model with which their effects can be quantitatively assessed.

Even though miss distance prediction in Region 2 requires the application of numerical methods because the overall problem has no closed form solution, once the MaRV and interceptor loci are glued together, the problem becomes formally tractable once again. The various derivatives obtained in the appendices can now be employed as sensitivity analysis and error propagation tools, if some mild assumptions are made about the nature of the input errors, as symbolized in Figure 3.12.

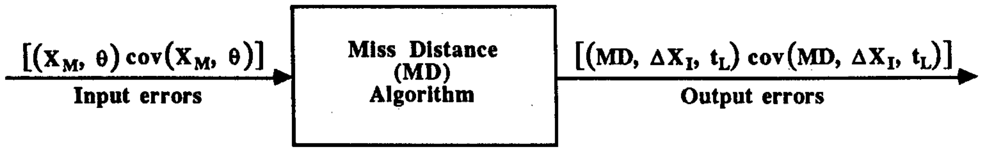

$[(X_M, \theta)\,\mathrm{cov}(X_M, \theta)]$
Input errors
Miss Distance (MD) Algorithm
$[(MD, \Delta X_I, t_L)\,\mathrm{cov}(MD, \Delta X_I, t_L)]$
Output errors

Figure 3.12. Algorithm MD computes the sensitivity of output variables to errors and uncertainties in input variables.

Consider the MaRV state X_M and the parameter vector θ corrupted by errors and uncertainty summarized by the covariance matrix $\mathrm{cov}(X_M, \theta)$. If we assume that errors are small, Gaussian, and additive, and that MD is linear for small input disturbances, the covariance of the Gaussian output random vector $(MD, \Delta X_I, t_L)$ is simply

$$\mathrm{cov}(MD, \Delta X_I, t_2) = J\ \mathrm{cov}(X_M, \theta) J^T \quad , \tag{3.28}$$

where

$$J = \frac{\partial(MD, \Delta X_I, t_2)}{\partial(X_M, \theta)} \quad ,$$

the Jacobian of the output with respect to the input of MD. Since the partial derivatives in J are available, the computation of the output error distribution is routine.

3.4.4. MISS DISTANCE MODULE FOR REGION R_2 (MODULE C)

Recall that a waiting time t_w falls in region 2 if it falls in no other region. In contrast to the other regions where miss distances are either very small or very large, miss distances in R_2 usually assume a continuum of values, ranging typically from a few feet to several thousand feet. Because this region is computationally complex, we do not actually compute the entire window W_2, but only find the optimal waiting time t_w^* and the minimum miss distances MD^* at that optimal time. This approach is perfect if the minimum is unique, since the window has zero width and t_w^* fully specifies this singleton-window. Sometimes, a flat spot in the *MD* function does exist, but this is not important in practice because such spots are typically very narrow. Exceptions usually arise in those cases where the interceptor totally dominates the MaRV, but in such cases the battle manager will have ample opportunities to neutralize the MaRV near that flat spot, or in the other opportunity windows which will likely arise. A simplified flowchart of Module C is illustrated in Figure 3.13, and we conclude this chapter with a discussion of all its blocks, as we did for the other modules.

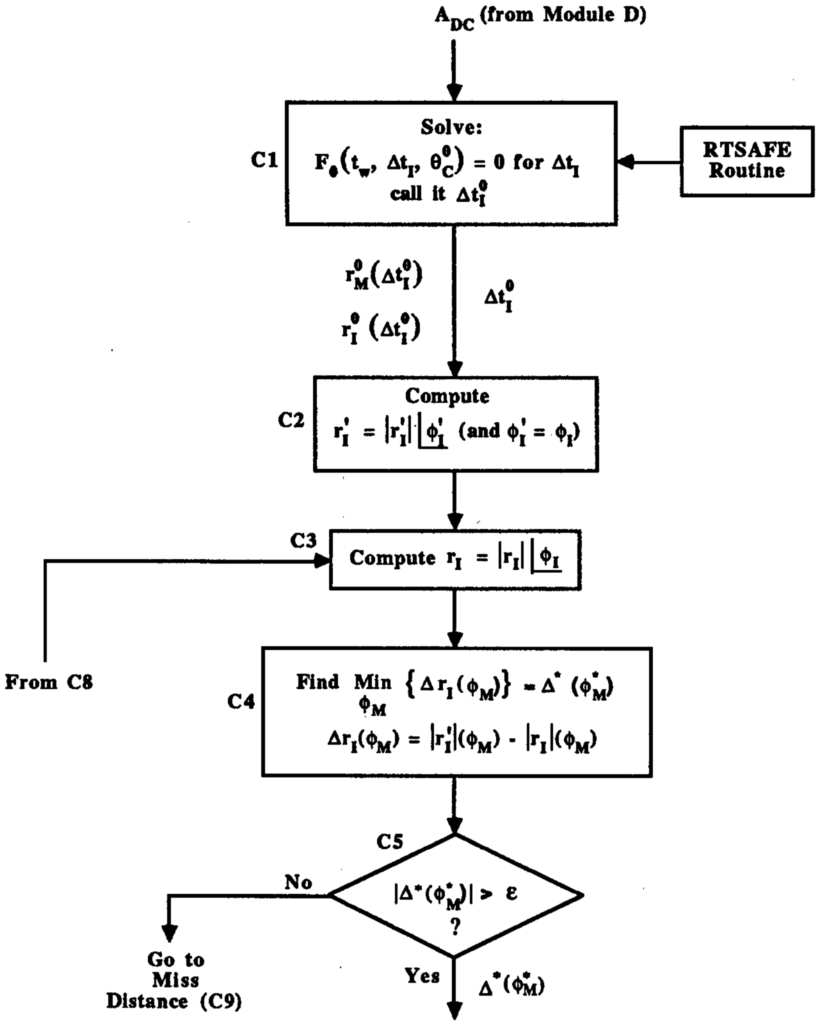
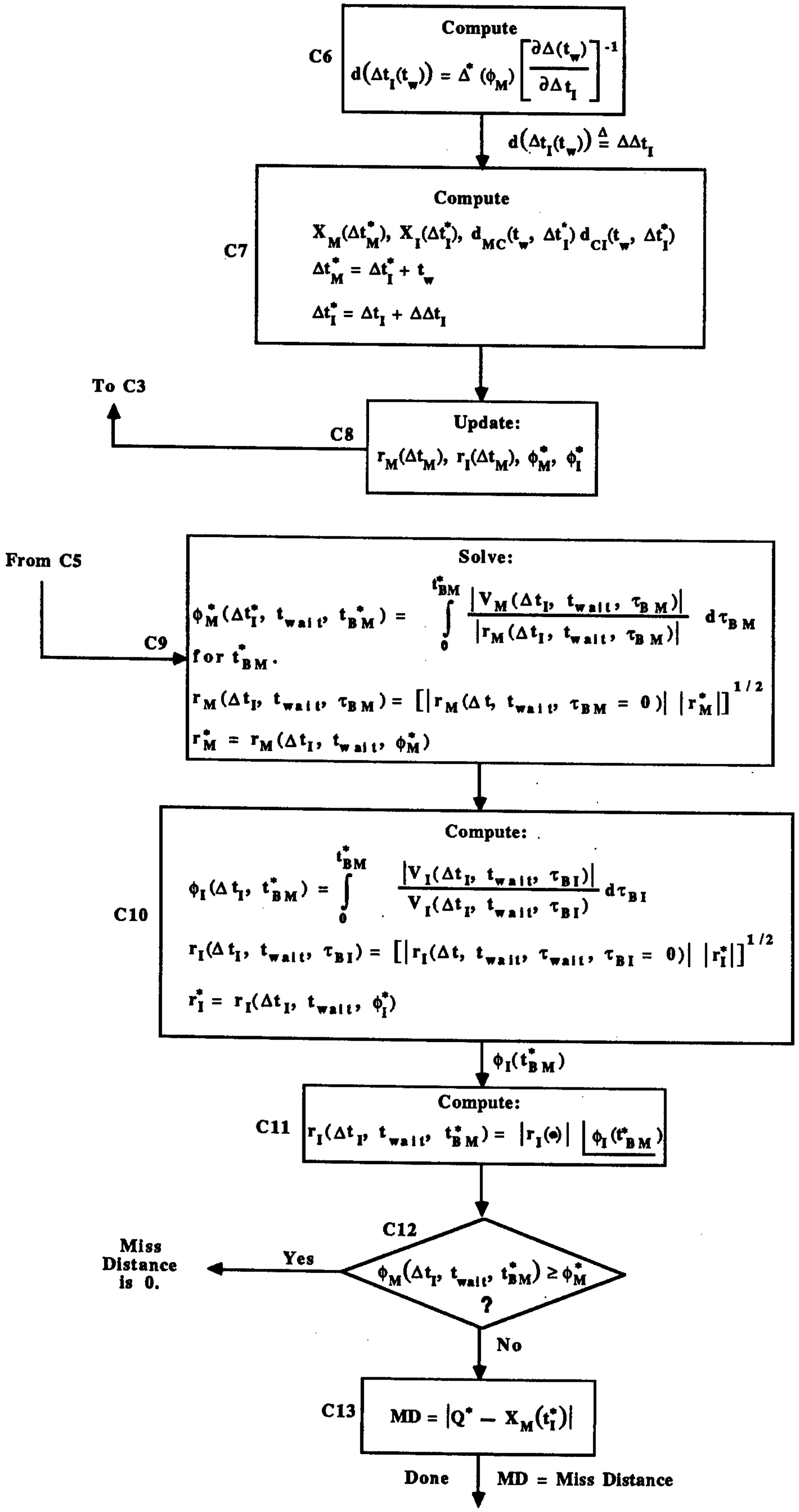

A_{DC} (from Module D)
C1
Solve:
$F_\theta(t_w, \Delta t_I, \theta_C^0) = 0$ for Δt_I
call it Δt_I^0
RTSAFE Routine
$r_M^0(\Delta t_I^0)$
$r_I^0(\Delta t_I^0)$
Δt_I^0
C2
Compute
$r_I' = |r_I'| \angle \phi_I'$ (and $\phi_I' = \phi_I$)
C3
Compute $r_I = |r_I| \angle \phi_I$
From C8
C4
Find $\underset{\phi_M}{\text{Min}} \{\Delta r_I(\phi_M)\} = \Delta^*(\phi_M^*)$
$\Delta r_I(\phi_M) = |r_I'|(\phi_M) - |r_I|(\phi_M)$
C5
$|\Delta^*(\phi_M^*)| > \varepsilon$
?
No
Go to Miss Distance (C9)
Yes
$\Delta^*(\phi_M^*)$

Figure 3.13. Simplified flowchart for Module C (continued on the following page).


C6
Compute
$d(\Delta t_I(t_w)) = \Delta^*(\phi_M)\left[\frac{\partial\Delta(t_w)}{\partial\Delta t_I}\right]^{-1}$
$d(\Delta t_I(t_w)) \triangleq \Delta\Delta t_I$
C7
Compute
$X_M(\Delta t_M^*),\ X_I(\Delta t_I^*),\ d_{MC}(t_w,\ \Delta t_I^*)\ d_{CI}(t_w,\ \Delta t_I^*)$
$\Delta t_M^* = \Delta t_I^* + t_w$
$\Delta t_I^* = \Delta t_I + \Delta\Delta t_I$
To C3
C8
Update:
$r_M(\Delta t_M),\ r_I(\Delta t_M),\ \phi_M^*,\ \phi_I^*$
From C5
C9
Solve:
$\phi_M^*(\Delta t_I^*,\ t_{wait},\ t_{BM}^*) = \int_0^{t_{BM}^*} \frac{|V_M(\Delta t_I,\ t_{wait},\ \tau_{BM})|}{|r_M(\Delta t_I,\ t_{wait},\ \tau_{BM})|}\ d\tau_{BM}$
for t_{BM}^*.
$r_M(\Delta t_I,\ t_{wait},\ \tau_{BM}) = [|r_M(\Delta t,\ t_{wait},\ \tau_{BM} = 0)|\ |r_M^*|]^{1/2}$
$r_M^* = r_M(\Delta t_I,\ t_{wait},\ \phi_M^*)$
C10
Compute:
$\phi_I(\Delta t_I,\ t_{BM}^*) = \int_0^{t_{BM}^*} \frac{|V_I(\Delta t_I,\ t_{wait},\ \tau_{BI})|}{V_I(\Delta t_I,\ t_{wait},\ \tau_{BI})}\ d\tau_{BI}$
$r_I(\Delta t_I,\ t_{wait},\ \tau_{BI}) = [|r_I(\Delta t,\ t_{wait},\ \tau_{wait},\ \tau_{BI} = 0)|\ |r_I^*|]^{1/2}$
$r_I^* = r_I(\Delta t_I,\ t_{wait},\ \phi_I^*)$
$\phi_I(t_{BM}^*)$
C11
Compute:
$r_I(\Delta t_I,\ t_{wait},\ t_{BM}^*) = |r_I(\bullet)|\ \underline{|\phi_I(t_{BM}^*)}$
C12
$\phi_M(\Delta t_I,\ t_{wait},\ t_{BM}^*) \geq \phi_M^*$
?
Yes
Miss Distance is 0.
No
C13
$MD = |Q^* - X_M(t_I^*)|$
Done
MD = Miss Distance

The algorithm starts with the initialization step discussed in Section 3.4.2 (C1), and finds the initial values for $r_I'(\phi_M)$ and $r_I(\phi_M)$ (C2, C3). Then the distance $\Delta^*(t_M)$ between the MaRV and interceptor loci is found in Block C4. If that distance exceeds the acceptance threshold ϵ (C5), a correction $\Delta\Delta t_I$ is computed in Block C6, all relevant variables are updated in C7 and C8, and a new distance Δ^* is computed in C3 and C4. Once the loci have been joined at Q^*, the banking time t^*_{BM} taken by M to reach Q^* from $X_M(\Delta t_M)$ (starting position for the turn) is calculated in C9. Then the angular distance $\phi_I(t^*_{BM})$ travelled by I in time t^*_{BM} is found in C10 and the corresponding value for the turn radius $r_I(\phi_I)$ is found in C11.

If $\phi_M(t^*_I)$ is less than the angular displacement required by M to reach Q^*, the miss distance is zero (C12).

If not, the miss distance equals

$$MD = |Q^* - X_M(t^*_I)| \quad ,$$

as discussed in the text (see Equation (3.6)).

Once the miss distance is calculated, the result is fed to the p_K module in the battle manager, as discussed in Chapter 1.

3.4.5. MAXIMIZING THE PROBABILITY OF KILL

Recall from Equation (3.1) that p_k depends not only on the miss distance, but also on the lethality radius of the interceptor. For some warheads, lethality itself depends upon atmospheric density, hence upon the altitude where the weapon is used. Minimizing *only* the miss distance will not necessarily yield a maximum p_k for such weapons, and the lethality function or kill radius must be inserted into the optimization loop, as shown in Figure 3.14, which is an extension of Figure 3.2. Accordingly, when considering such systems, the flowchart of Figure 3.13 is modified to output p_k values instead of miss distance values.

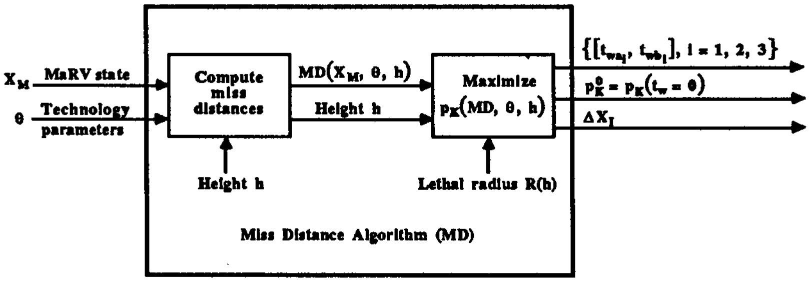
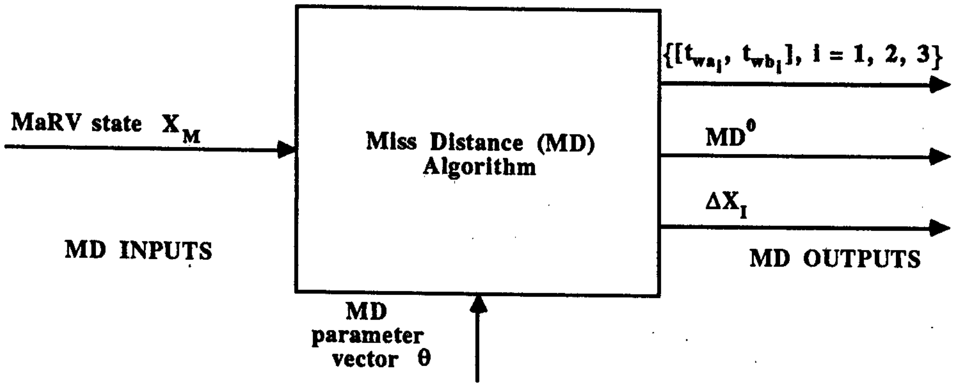


Figure 3.14. For interceptors whose lethality depend upon height (h), the MD algorithm includes the lethal radius ($R(h)$) in the optimization loop.

Chapter 4

EXPERIMENTAL RESULTS

The MD Algorithm was programmed in VAX Fortran to allow comprehensive parametric investigations, and in this chapter we report the results of various computational runs. The input-output diagram of Figure 4.1 illustrates all the user-adjustable parameters of MD. Of these, only a few key parameters are explored in this chapter (refer to the text to review how these are formally related).


MaRV state X_M
MD INPUTS
Miss Distance (MD) Algorithm
$\{[t_{wa_i}, t_{wb_i}], i = 1, 2, 3\}$
MD^0
ΔX_I
MD OUTPUTS
MD parameter vector θ

$\theta =$

H_0:	Maneuver/atmospheric boundary
h_{k0}:	Keep-out zone altitude
$\{T_{I_\alpha}\}$:	Interceptor thrust parameter for type α.
α:	Two interceptor types, $\alpha = 1$ and $\alpha = 2$
k_{DI_α}:	Interceptor drag constants
k_{DM}:	MaRV drag constant
m_M, m_{I_α}:	MaRV and interceptor masses
C_{LM}, C_{LI_α}:	Lift coefficients
S_M, S_I:	Effective vehicle area
ρ:	Atmospheric density
δ:	Interceptor system time delay
$\mathrm{cov}(x_M, \theta)$:	Covariance of MaRV state and parameter vector
$R(\gamma, \beta, \theta)$:	Orientation of MaRV maneuvering plane
$X^0_{I_\alpha}$:	Interceptor location
Various numerical parameters	

Figure 4.1. Several model parameters are accessible by the user.

The parameters discussed in the various sections of this chapter are:

4.1. H_0: Atmospheric or maneuver boundary.

4.2. h_{ko}: Keep-out zone altitude.

4.3. α: Interceptor type.

4.4. m_{I_α}: Interceptor mass.

4.5. δ: Interceptor system time delay.

4.6. X_I^0: Interceptor location.

4.7. V_M^0: MaRV speed.

Although the experimental runs were made by varying only one parameter at a time, significant effort was made to fix the input vector X_M and the remaining parameters at those values most likely to reveal important sensitivities and parameter dependencies. Additional efforts were made in some cases, particularly in Section 4.1 below, to choose parameter values for which three opportunity windows exist. This was done essentially to reduce the number of computer runs, and has no particular physical significance. As we shall see, a typical setting for X_M^0 was (−728, 287, 10), expressed as a vector in units of thousands of feet (kft). The MaRV aimpoint, determined by X_M^0 and the initial velocity vector V_M^0, was (125, 0, 0) kft, and the initial speed $|V_M^0|$ was 22 kft/sec.

The final speed $|V_M^F|$ was 2630 ft/sec, nominal MaRV mass was 15.25 slugs, the keep-out zone altitude was 5000 ft, the nominal interceptor (Type 1) mass was 33.59 slugs, the maneuvering limit (H_0) was 70 kft and the system time delay (δ) was 0.2 sec. No statistical errors were propagated in our tests, and the covariance matrix cov(X_M, θ) was not used in our deterministic analysis.

To properly interpret all the plots in this chapter, recall that an "opportunity window" is a subset of waiting times t_w at which the miss distance achieves a minimum. Since our miss distance predictions are rooted in the most recent update of the MaRV state X_M, such windows are computed for each such update at a rate consistent with the battle manager's "real-time" clock, whose increment is typically 0.1 sec. Observe also that the occurrence of triple windows is rare. Such a situation typically implies that the MaRV altitude is large (in fact exceeds H_0), and that it is *currently* aimed overhead from the interceptor's point of view. We shall see that, as the MaRV approaches its target, Window 1 disappears as X_M enters the maneuvering region, and Window 3 disappears because tailchasing becomes increasingly difficult, unless the separation between the interceptor base and the MaRV's target is large.

Comparing the results in this chapter with the work of others [13,14–18] we find general agreement in the conclusions, but have also discovered some omissions in our work, particularly with reference to the "field of view" of both vehicles. When interceptors are equipped with internal homing devices, or MaRVs with internal missile avoidance devices, limitations in the field of view can cause severe errors in their range measurements. Such errors can severely distort miss distance predictions [14,15].

4.1. Effects of Varying the Atmospheric/Maneuvering Boundary H_0

Figure 4.2 illustrates a particular situation for which the current state estimate X_M produces three windows. Window 1 extends roughly from 0 to 4 sec and indicates that, if an interceptor is launched *now* ($t_w = 0$), the miss distance will be very small; essentially the same as if the MaRV were in fact a Ballistic Re-entry Vehicle (BRV), since maneuvering is practically excluded in Window 1. The width of the window (about 4 sec) shows that the predicted intercept point is well above H_0 and that interception would still occur above H_0 even if interceptor launch took place 4 sec later. In any case, the best strategy is to launch at the earliest time at which Window 1 opens.

Even though the existence of two additional windows is mostly of theoretical interest, since the policy is to launch as soon as the first window opens, observe that a 2-sec tailchase window (Window 3) exists. This usually indicates, based on the most recent state observation, that the MaRV is headed overhead towards a target significantly far away from the interceptor site. Also note that, if no interceptor were fired for some reason, the miss distance would be about 3500 ft if the interceptor were launched at the optimal time in Region 2 ($t_w^* \approx 31$ sec).

When the maneuvering or atmospheric ceiling is lowered from 100 kft to 70 kft (Figure 4.3), Window 1 is significantly wider, as would be expected, since the MaRV behaves ballistically for a considerably larger period of time. Observe also that the tailchase window is much narrower, down to about 0.25 sec. This is because the MaRV starts decelerating much later in its trajectory thereby maintaining its initial speed much longer, and thus winning the tailchase during the first 2 sec of the tailchase region (from $t_w = 29$ sec to $t_w = 31$ sec).

4.2. Varying the Keep-Out Zone

The keep-out zone can significantly affect the minimum miss distance achievable, for two major reasons. The first is associated with the type of interceptor used, and will be discussed in the next section. The second arises in the tailchase region (Window 3). Whenever Window 3 is open, reducing the keep-out height h_{ko} provides increased

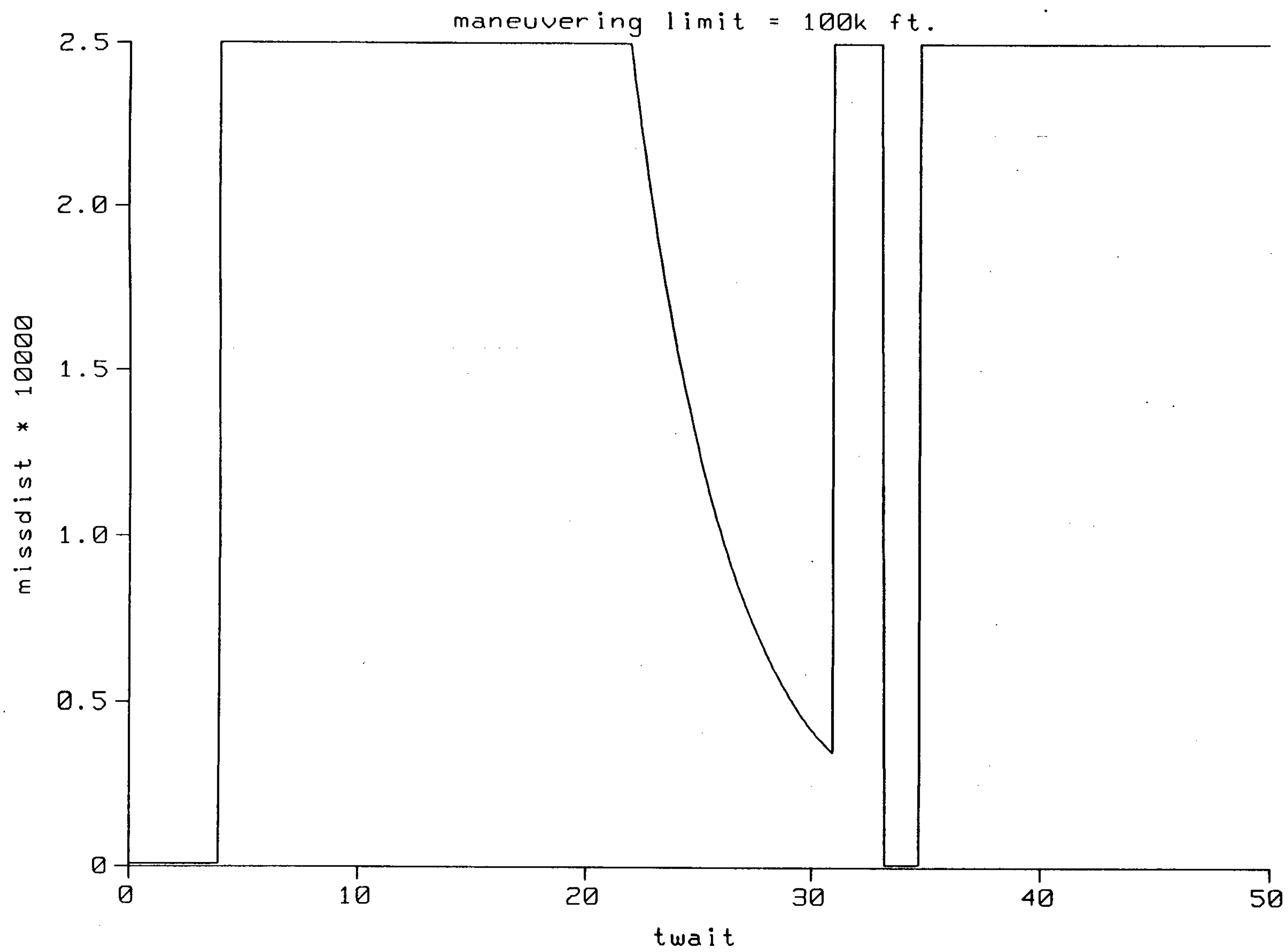


Figure 4.2. Opportunity windows when H_0 is 100 kft (t_{wait} in sec, and miss distances in 10^4 ft). $X_M^0 = (-728\text{k}, 287\text{k}, 10\text{k})$, aimpoint $= (125\text{k}, 0, 0)$, $V_M^0 = 22$ kfps, $V_{M_{\text{final}}} = 2630$ fps, MaRV mass = 15.25 slugs, $X_i^0 = (-20\text{k}, 0, 0)$, Type 1, int. mass = 33.59 slugs, $h_{ko} = 5$ kft, $\delta = 0.2$ s.

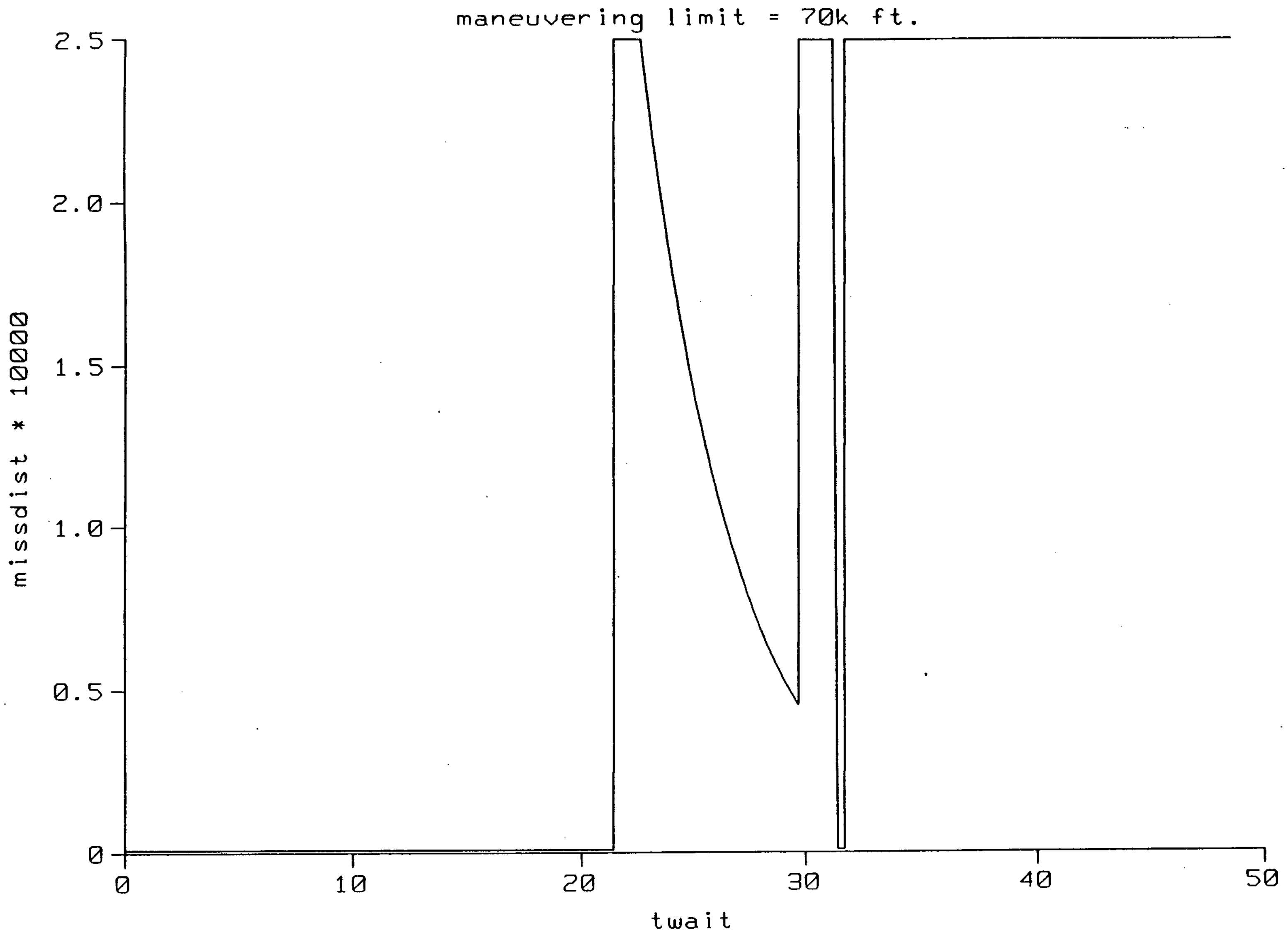


Figure 4.3. Opportunity windows when $H_0 = 70$ kft (t_{wait} in sec and distances in 10^4 ft). $X_M^0 = (-728\text{k}, 287\text{k}, 10\text{k})$, aim point $= (125\text{k}, 0, 0)$, $V_M^0 = 22$ kfps, $V_{M_{\text{final}}} =$ 2630 fps, MaRV mass = 15.25 slugs, $X_i^0 = (-20\text{k}, 0, 0)$, Type 1, int. mass = 33.59 slugs, $h_{ko} = 5$ kft, $\delta = 0.2$ s.

if h_{ko} is opportunity for the interceptor since tailchasing typically arises at low altitudes where the MaRV speed is relatively low and decreasing, and the interceptor's speed is still increasing. Even reduced by only a few thousand feet, the effect is quite noticeable, as evidenced by Figures 4.4 and 4.5, where a window width increase of about 1 sec may be observed. We emphasize that Window 3 closes, not because the MaRV wins the tailchase once again, but only because the engagement—more exactly the predicted intercept point X_C—occurs *inside* the keep-out zone.

The effects of varying h_{ko} also show clearly in Region 2, as shown in Figures 4.6 and 4.7. When h_{ko} is 5000 ft, the minimum miss distance $MD(t_w^*)$ is about 1000 ft, whereas $MD(t_w^*)$ is only 450 ft when h_{ko} is lowered to 1000 ft. This is mostly due to the fact that the MaRV speed is lower at intercept, and this is important considering the interceptor time delay of 0.2 sec used in this example.

4.3. Different Interceptor Types

Recall from Chapter 2 that a Type 1 interceptor is designed to have a long reach and is intended for "overlay interception." In contrast, a Type 2 interceptor typically produces higher thrust, but for a shorter period of time, thereby reaching a given speed considerably earlier than a Type 1. It may thus be considered a "last ditch defense" vehicle intended for "underlay interception" near the keep-out zone.

Our two interceptor types are sufficiently different in that respect that it was simple to construct a scenario where, due to the keep-out zone constraint, a Type 1 vehicle was unable to intercept a MaRV, whereas a Type 2 had a Window 2 extending over several sec.

Referring to Figure 4.8, the miss distance remains at 10,000 ft for the Type 1, whereas it drops to a minimum of 1200 ft in Region 2 for a Type 2.

4.4. Varying Interceptor Mass

Because the minimum radius of curvature of a maneuver is directly proportional to the mass of a vehicle (see Chapter 2), and the path length along the maneuver is also proportional to that radius (see Chapter 3), one would expect that the miss distance would also be roughly proportional to interceptor mass. This is in fact typically the case, as shown in Figure 4.9, where we varied the interceptor mass from 3.36 slugs to 335.9 slugs. A converse relationship obviously holds for the MaRV. As its weight is increased, the miss distance decreases since its maneuvering capability decreases proportionately, but this test was not executed on the computer.

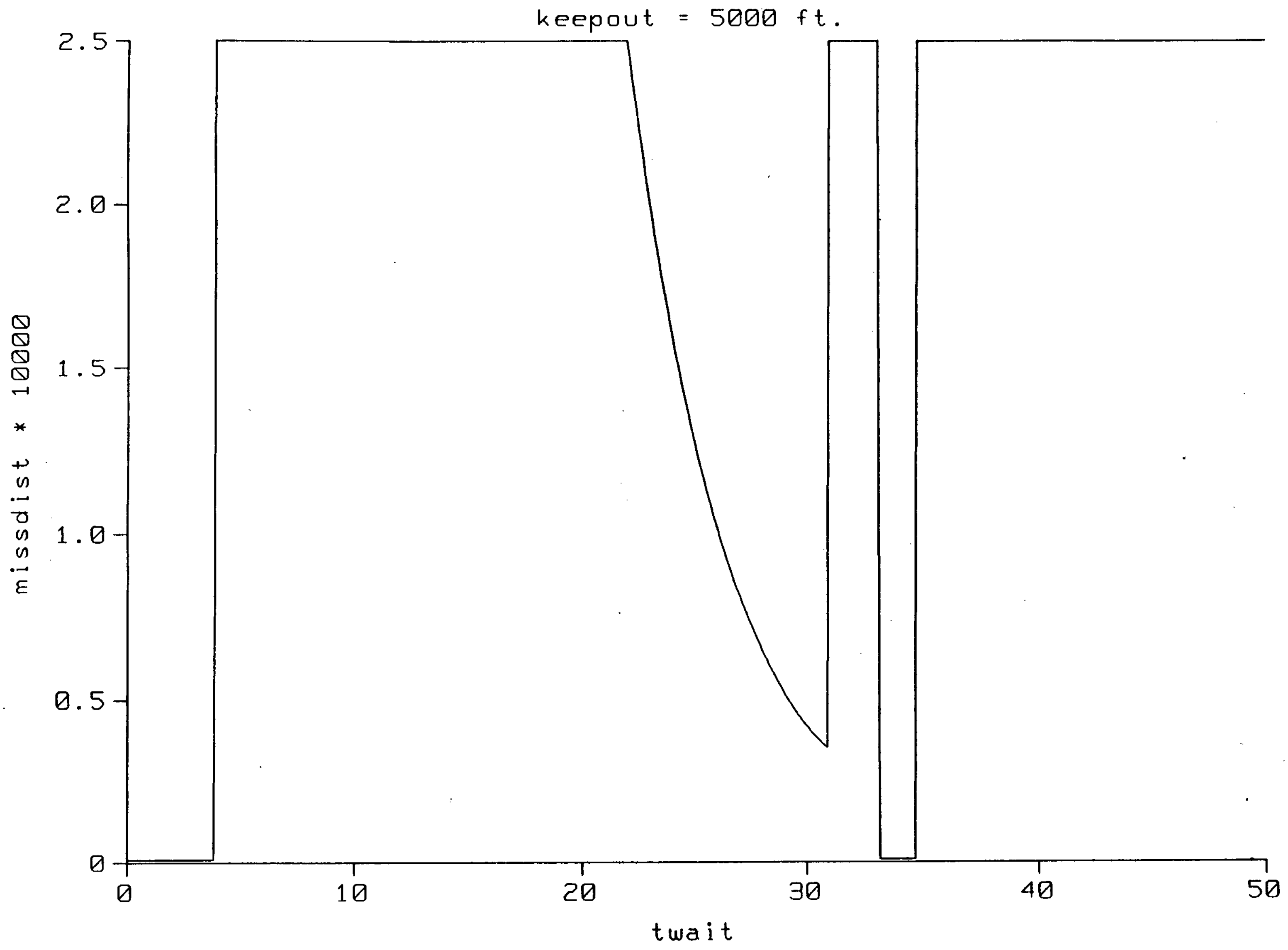


Figure 4.4. The three windows when the keep-out height h_{ko} is 5000 ft (t_{wait} in sec and distances in 10^3 ft).
$X_M^0 = (-728\text{k}, 287\text{k}, 10\text{k})$, aim point = (125k, 0, 0), $V_M^0 = 22$ kfps, $V_{M_{\text{final}}} = 2630$ fps, MaRV mass = 15.25 slugs, $X_I^0 = (-20\text{k}, 0, 0)$, Type 1, int. mass = 33.59 slugs, $H_0 = 100$ kft, $\delta = 0.2$ s.

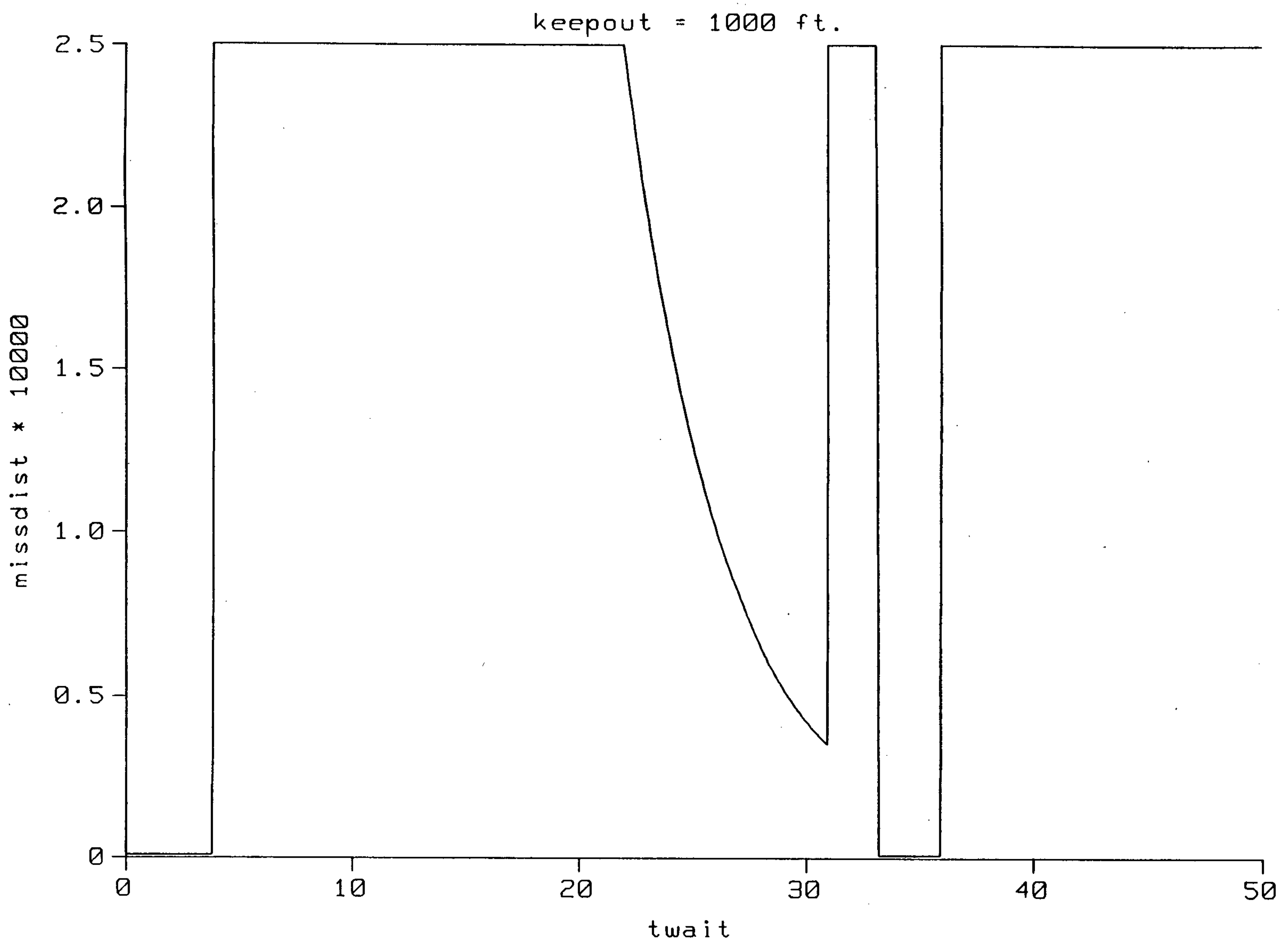


Figure 4.5. The three windows when the keep-out height is h_{ko} is 1000 ft (t_{wait} in sec and distances in 10^4 ft).
$X_M^0 = (-728\text{k},\ 287\text{k},\ 10\text{k})$, aim point $= (125\text{k},\ 0,\ 0)$, $V_M^0 = 22$ kfps, $V_{M_{\text{final}}} = 2630$ fps, MaRV mass $= 15.25$ slugs, $X_I^0 = (-20\text{k},\ 0,\ 0)$, Type 1, int. mass $= 33.59$ slugs, $H_0 = 100$ kft, $\delta = 0.2$ s.

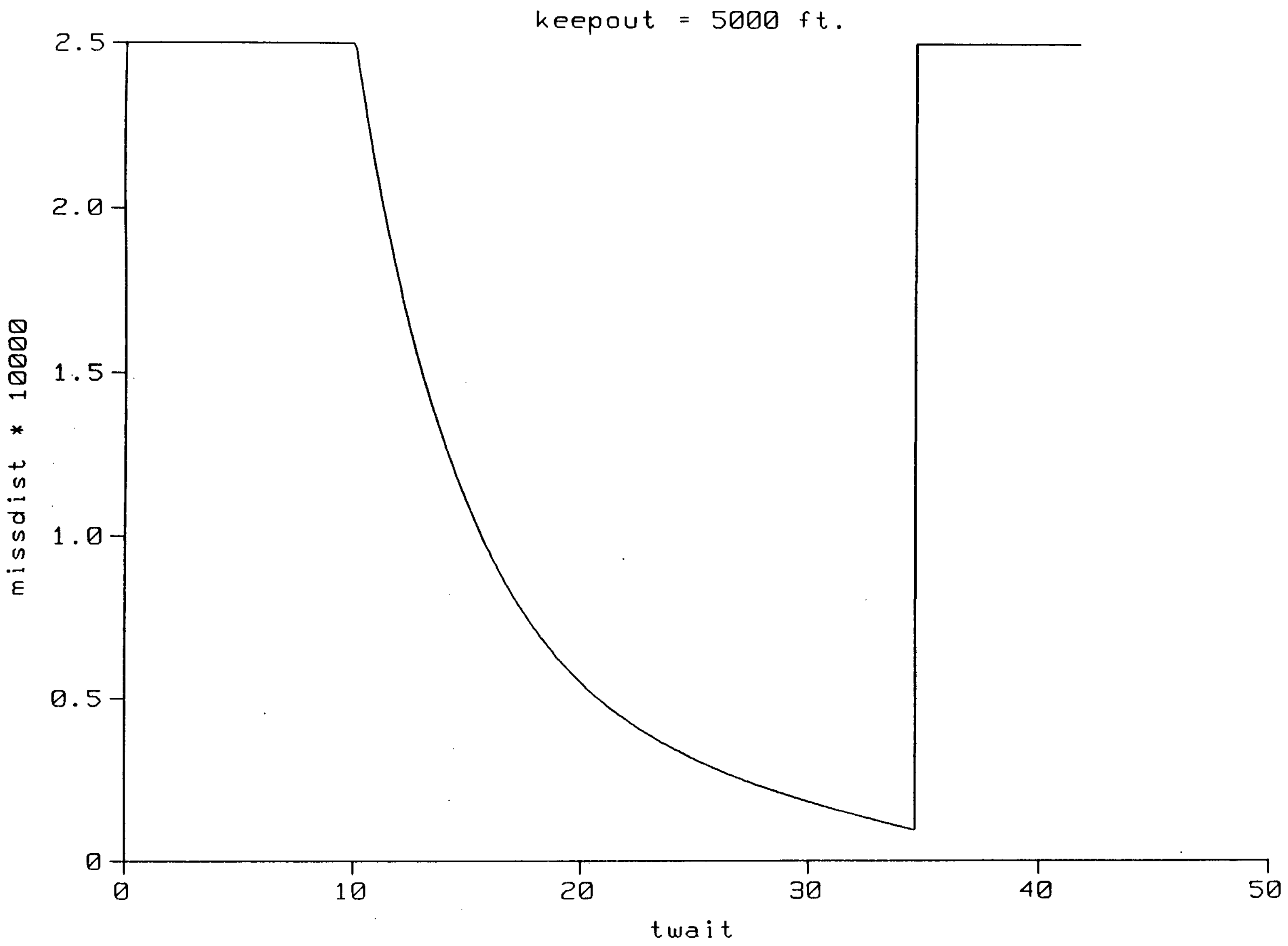


Figure 4.6. Window 2 when h_{ko} is 5000 ft (t_{wait} in sec and distances in 10^4 ft). $X_M^0 = (-728\text{k}, 287\text{k}, 10\text{k})$, aim point $= (-10\text{k}, 0, 0)$, $V_M^0 = 22$ kfps, $V_{M_{\text{final}}} = 2630$ fps, MaRV mass $= 15.25$ slugs, $X_I^0 = (5\text{k}, 0, 0)$, Type 1, int. mass $= 33.59$ slugs, $H_0 = 70$ kft, $\delta = 0.2$ s.

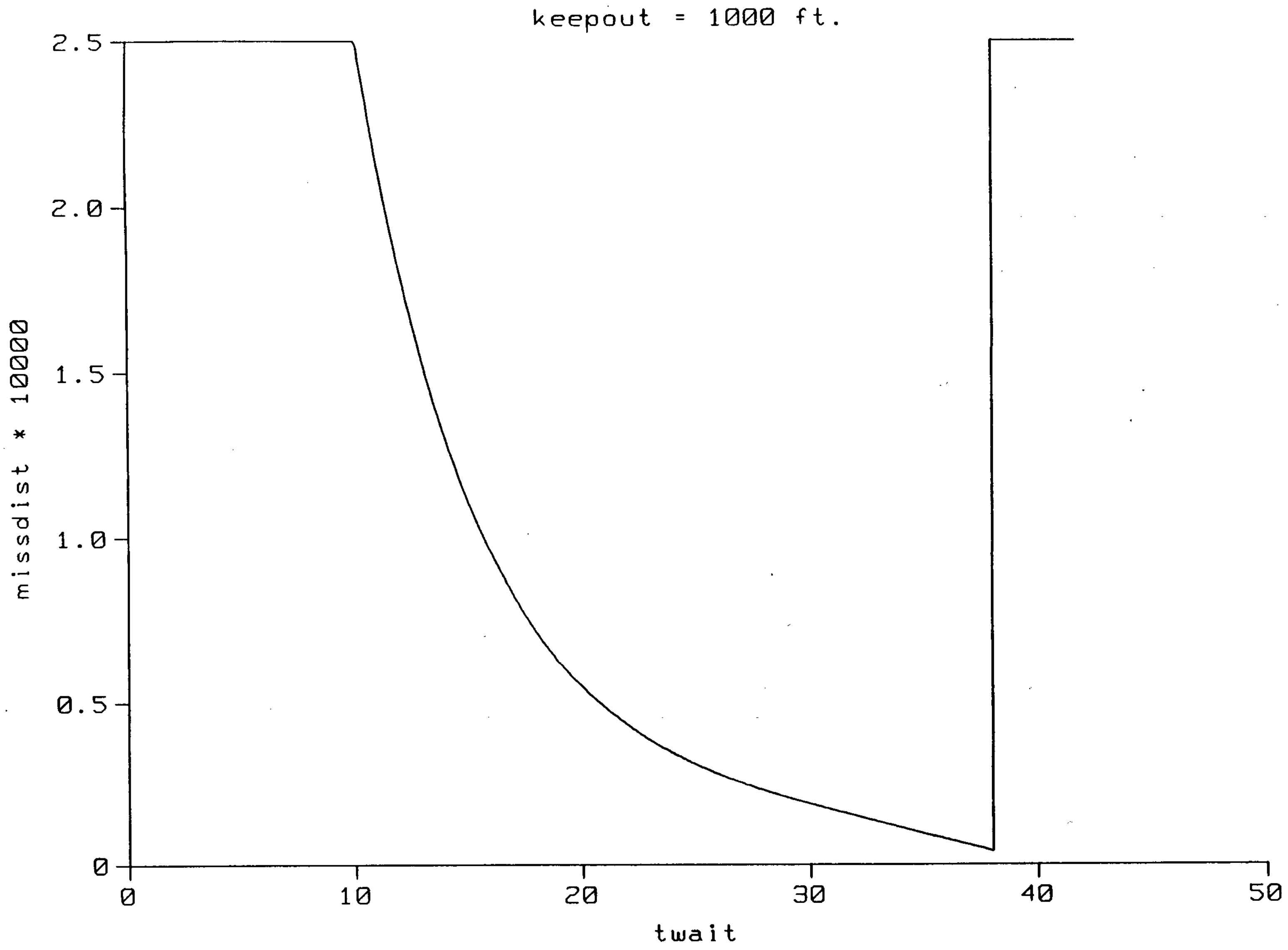


Figure 4.7. Window 2 when h_{ko} is 1000 ft (t_{wait} in sec and distances in 10^4 ft). X_M^0 = (−728k, 287k, 10k), aim point = (−10k, 0, 0), V_M^0 = 22 kfps, $V_{M_{\text{final}}}$ = 2630 fps, MaRV mass = 15.25 slugs, X_I^0 = (5k, 0, 0), Type 1, int. mass = 33.59 slugs, H_0 = 70 kft, δ = 0.2 s.

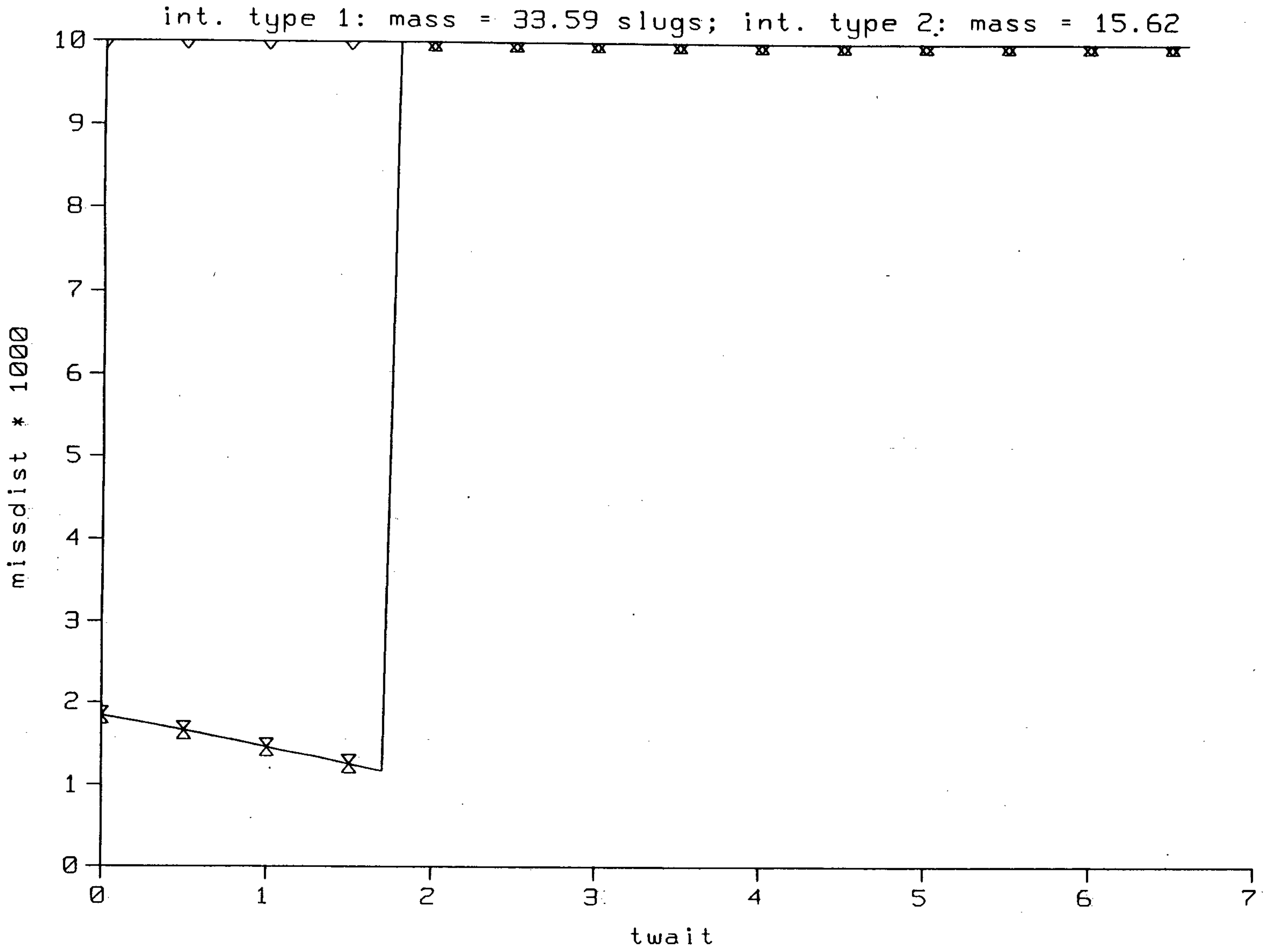


Figure 4.8. Case where a typical Type I interceptor misses the MaRV near the keepout zone, but a Type II kills the MaRV (miss distance in kft, t_{wait} in sec). $X_M^0 = (-65\text{k}, 50\text{k}, 2\text{k})$, aim point $= (0, 0, 0)$, $V_M^0 = 22$ kfps, $V_{M_{\text{final}}} = 2600$ fps, MaRV mass $= 15.25$ slugs, $h_{ko} = 5000$ ft, $X_I^0 = (40\text{k}, 0, 10\text{k})$, $H_0 =$ 70 kft, $\delta = 0.2$ s.

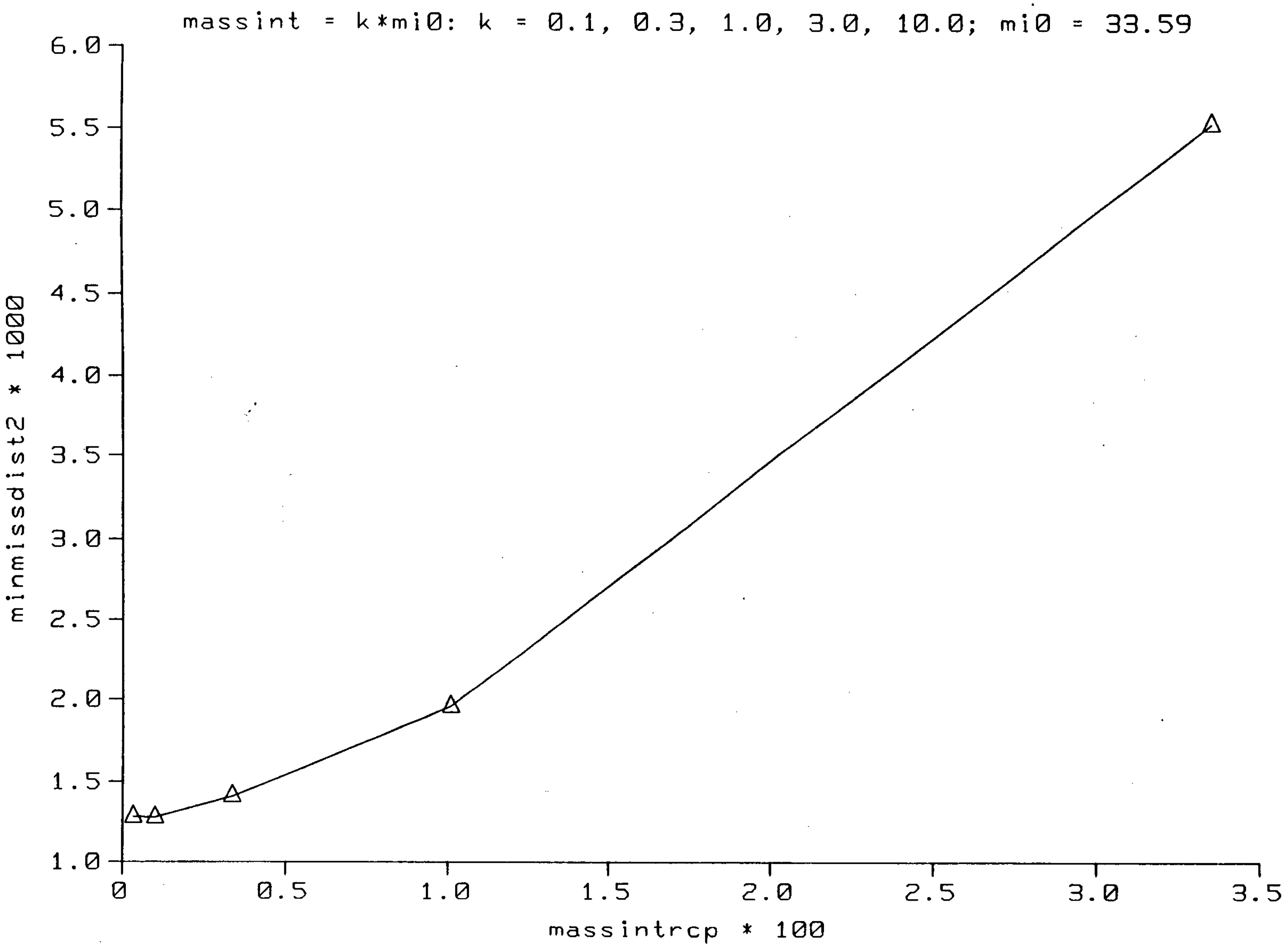


Figure 4.9. The minimum miss distance in Region 2 is very sensitive to interceptor terminal mass (t_{wait} in sec and distances in 10^3 ft).
X_M^0 = (−400k, 150k, −50k), aim point = (0, 0, 0), V_M^0 = 22 kfps, $V_{M_{\text{final}}}$ = 5000 fps, MaRV mass = 15.25 slugs, h_{ko} = 5000 ft, X_I^0 = (0, 0, 10k), type 1 interceptor, H_0 = 70 kft, δ = 0.2 s.

4.5. Accounting for Interceptor System Time Delay

Recently, the MD Algorithm has been augmented with the capability to account for interceptor time delays, as described in Sections 3.4.1 and 3.4.2. These delays can severely reduce interceptor performance, particularly if they exceed 0.2 sec. Figure 4.10, for instance, illustrates the miss distance in Region 2. As the time delay δ is increased from 0 to 0.5 sec, the minimum miss distance varies from 0 to about 4000 ft; a significant spread. For practical systems, δ ranges between 0.1 and 0.2 sec, and the corresponding miss distance ranges between 750 and 1500 ft.

To show the influence on the kill probability P_k, a lethality table was folded in, and the resulting P_k values were plotted in Figure 4.11. For this particular payload lethality assumption, a delay of up to 0.2 sec is quite tolerable, but performance drags off sharply as δ is increased beyond that value.

4.6. Interceptor Location

Interceptor base location strongly influences interceptor performance only when the interceptor is launched far from the MaRV aimpoint or destination. In two extreme cases shown in Figure 4.12, where the most recent MaRV position estimate was (-728 kft, 287 kft, 10 kft) and the interceptor was located at (-400 kft, 0, 0) and (+400 kft, 0, 0), the interceptor missed the MaRV because it was unreachable in both cases. In the first case, and even though the interceptor was located 400 kft towards the MaRV from the origin, the MaRV aimpoint was so far beyond the origin that it was totally unreachable, passing overhead. In the second case, every point along the MaRV trajectory was unreachable.

When the interceptor was moved to (-100 kft, 0, 0), the interceptor acquired a large exoatmospheric window of about 20 sec during which interception was assured.

When the interceptor was moved to the origin, additional windows opened up, and even a tailchase opportunity showed up, as seen from the top right diagram in Figure 4.12.

When the interceptor was located at (-100 kft, 0, 0), a wide exoatmospheric window appeared, as shown in the bottom left diagram. When located at (+100 kft, 0, 0), there was no Window 1, but a good opportunity in Region 2 appeared, together with a narrow tailchase window where the interceptor had the advantage. This window was essentially due to the fact that the MaRV was predicted to pass overhead, towards its extreme down range destination of (125 kft, 0, 0).

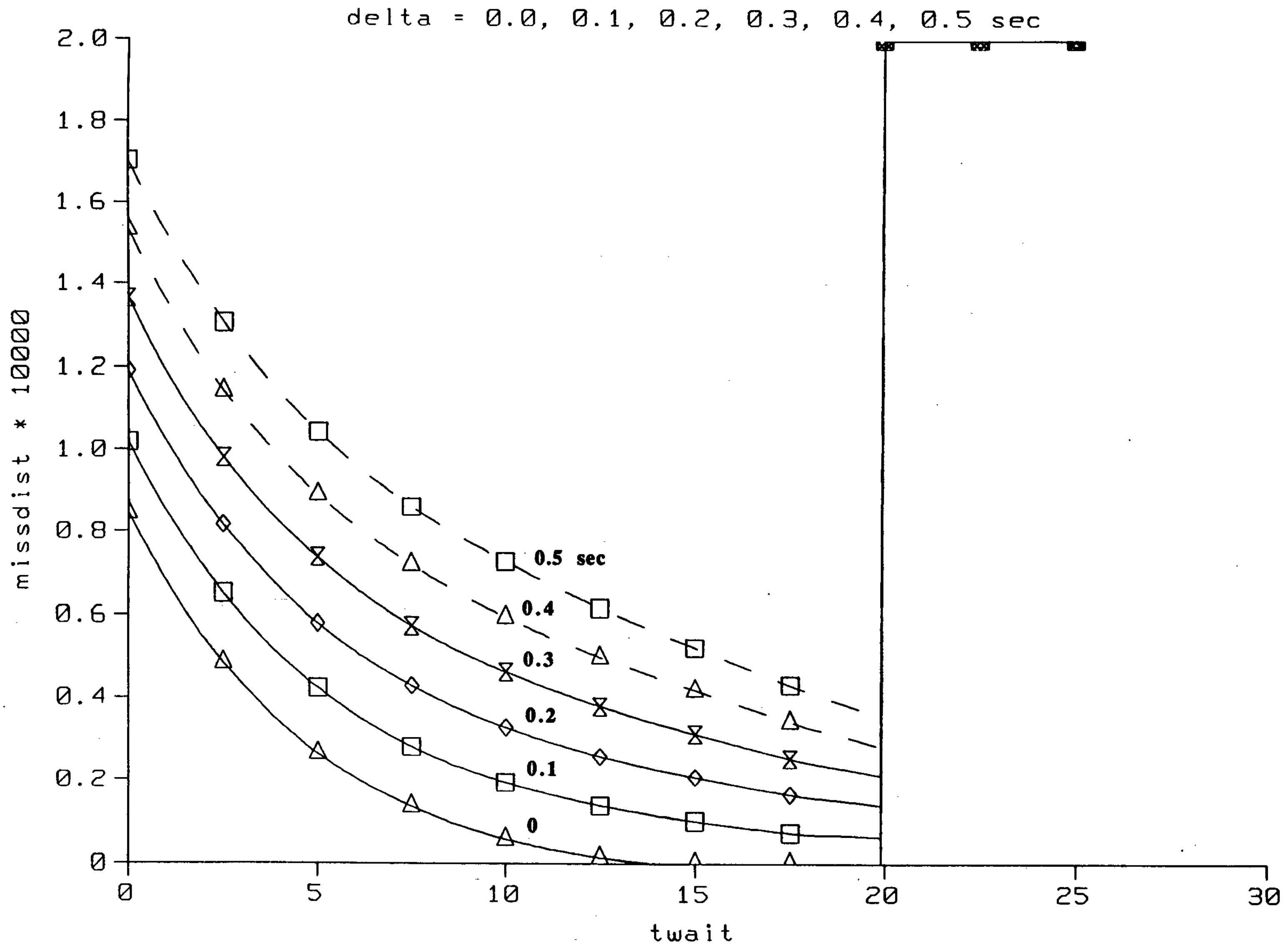


Figure 4.10. Interceptor time delay δ can significantly influence the miss distance (t_{wait} in sec, and miss distances in 10^4 ft). $X_M^0 = (-400\text{k}, 150\text{k}, -50\text{k})$, aim point = (0, 0, 0), $V_M^0 = 22$ kfps, $V_{M_{\text{final}}} = 5000$ fps, MaRV mass = 15.25 slugs, $h_{ko} = 5000$ ft, $X_I^0 = (0, 0, 10\text{k})$, type 1, int. mass = 33.59 slugs, $H_0 = 70$ kft.

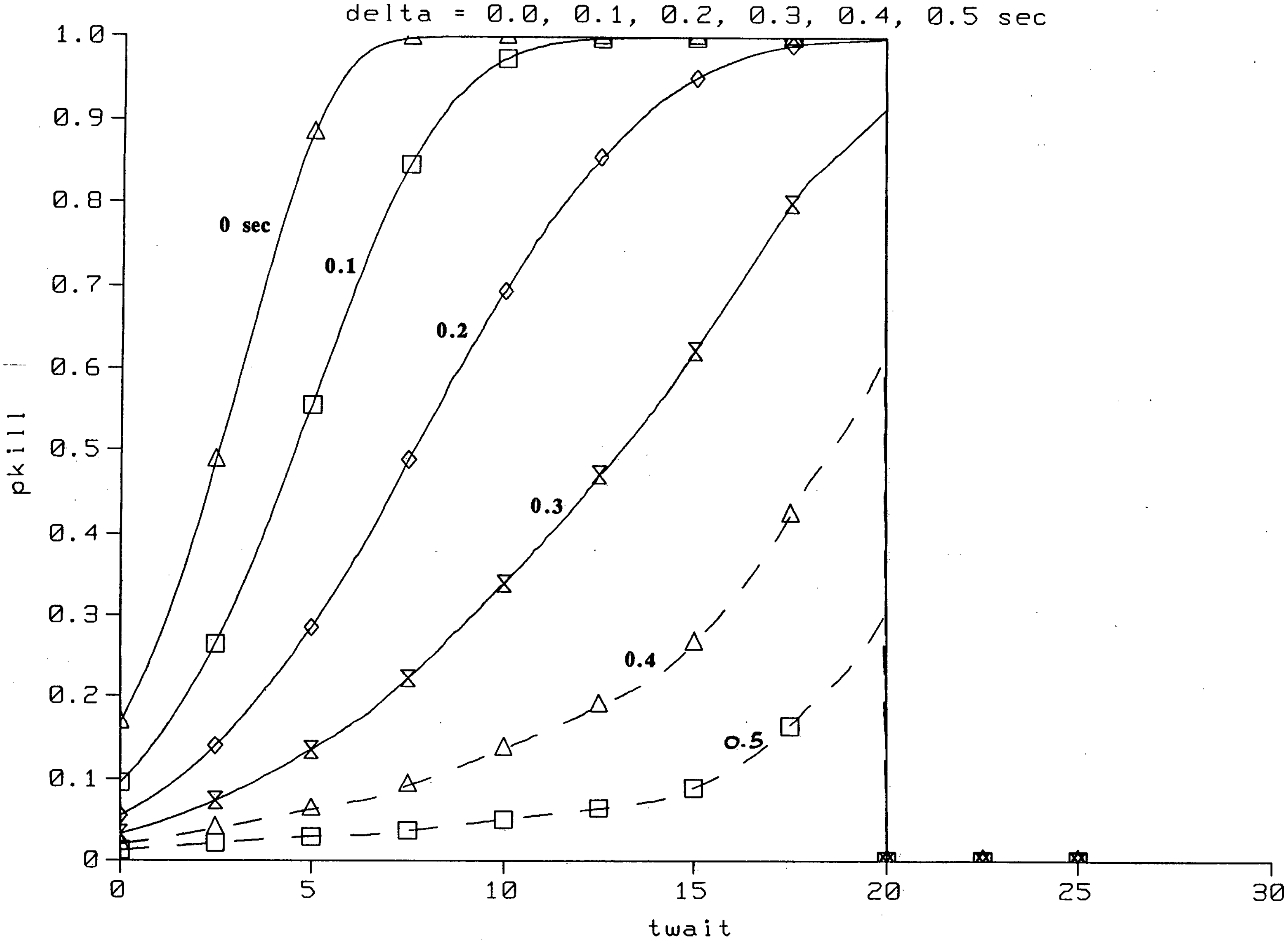


Figure 4.11. The prob(kill) is sensitive to interceptor time delays. $X_M^0 = (-400k, 150k, -50k)$, aim point $= (0, 0, 0)$, $V_M^0 = 22$ kfps, $V_{M_{\text{final}}} = 5000$ fps, MaRV mass $= 15.25$ slugs, $h_{ko} = 5000$ ft, $X_I^0 = (0, 0, 10k)$, type 1, int. mass $= 33.59$ slugs, $H_0 = 70$ kft.

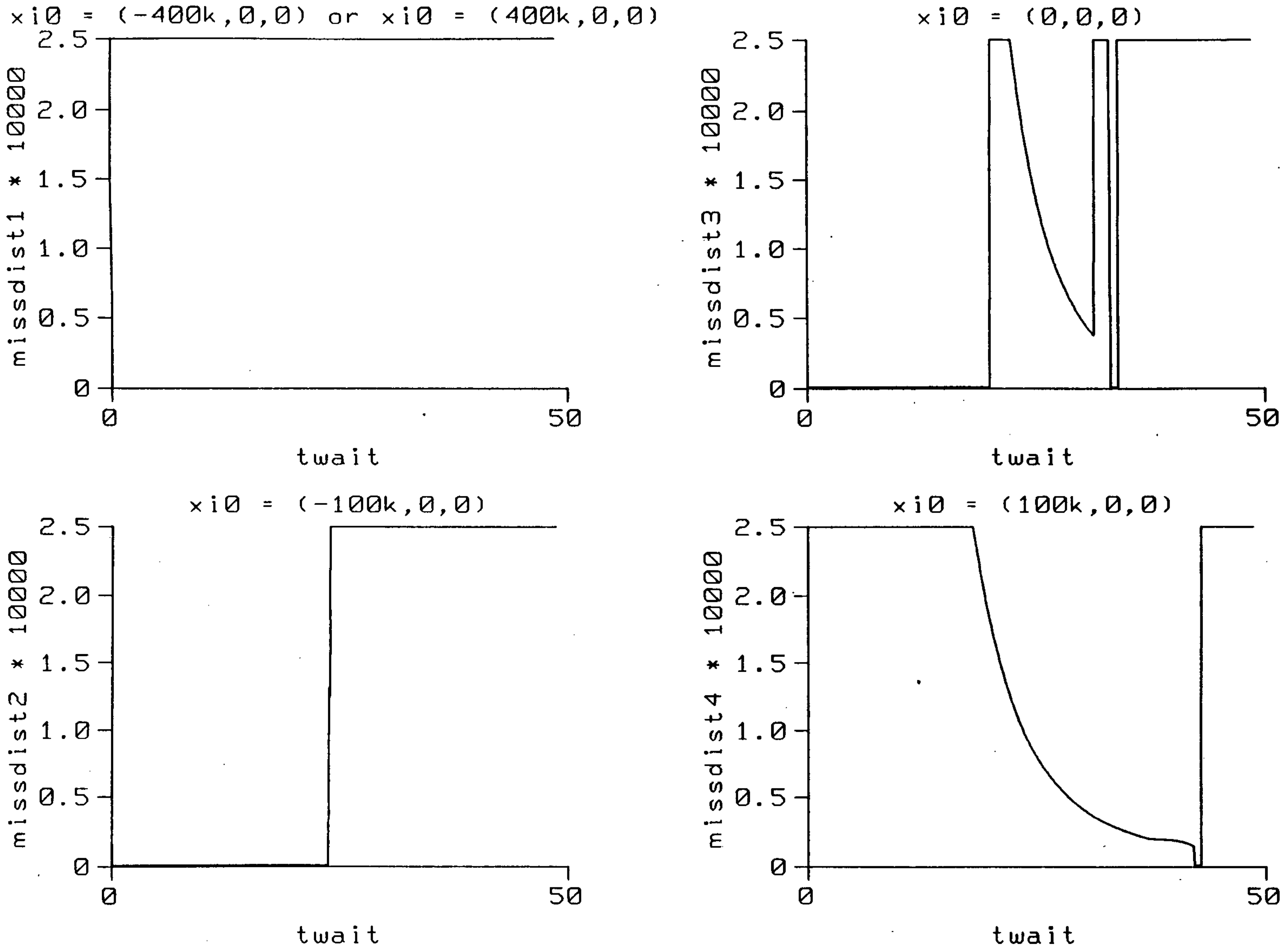


Figure 4.12. If they are based far from the MaRV aimpoint or destination, interceptors will often miss the MaRV entirely.
X_M^0 = (−728k, 287k, 10k), aim point = (125k, 0, 0), V_M^0 = 22 kfps, $V_{M_{\text{final}}}$ = 2630 fps, MaRV mass = 15.25 slugs, h_{ko} = 5000 ft, type 1, int. mass = 33.59 slugs, H_0 = 70 kft, δ = 0.2 s.

4.7. Varying the Initial MaRV Speed

The last parameter varied in our study was the MaRV speed $|V_M^0|$. Recall that this is the most recent speed estimate handed over by sensors or radars.

As seen from Figure 4.13, the miss distance is not as sensitive to MaRV speed changes as it is to mass changes, and the same comment applies to interceptor speeds. Consequently, one might be tempted to trade-off speed for mass, but be aware that the reduction in payload lethality associated with a reduction in payload mass could entirely offset any maneuvering advantage acquired. In some cases a reduction in payload mass may in fact reduce interceptor performance, but this possibility is strongly warhead-dependent.

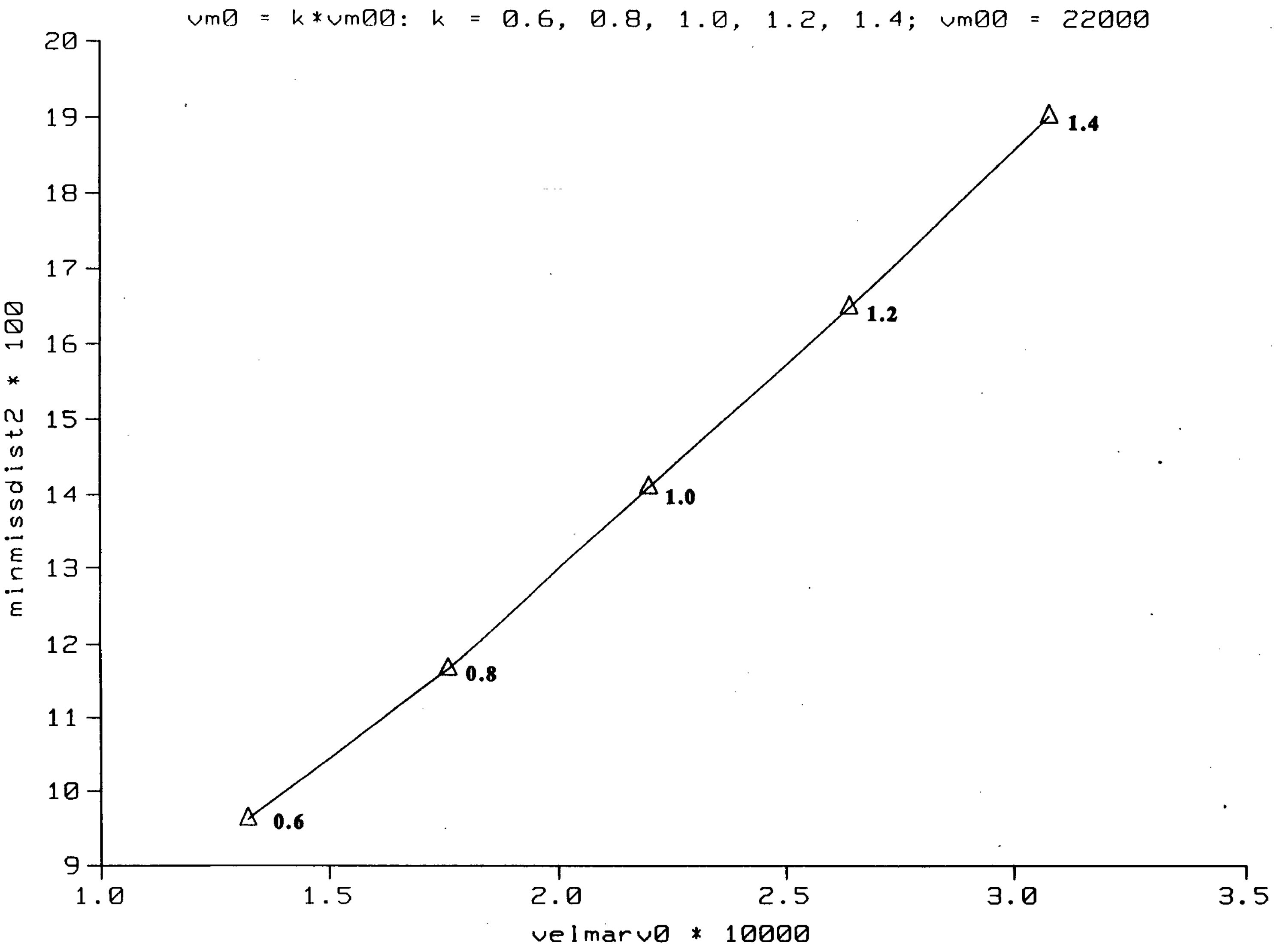


Figure 4.13. The miss distance (in 10^2 ft) is not very sensitive to initial MaRV speed (in 10^4 ft/sec).
$X_M^0 = (-400\text{k}, 150\text{k}, -50\text{k})$, aim point = (0, 0, 0), $V_{M_{\text{final}}} = 5000$ fps, MaRV mass = 15.25 slugs, $h_{ko} = 5000$ ft, int. mass = 33.59 slugs, $X_I^0 = (0, 0, 10\text{k})$, type 1 interceptor, $H_0 = 70$ kft $\delta = 0.2$ s.

Chapter 5

CONCLUSIONS AND FUTURE WORK

In allocating interceptors to decoyed MaRVs and BRVs, the battle manager must carefully process the discrimination information handed down by various sensors, and must be able to predict—or at least to estimate—the outcome of an engagement, sometimes far in advance of launch time. The battle manager discussed in this report satisfies such requirements. Discrimination information is processed in the form of a conditional probability that the threat considered is in fact a MaRV or a BRV, conditional to the observations, and this information itself conditions all the major decisions made in the battle manager.

The outcome of an engagement is judged by the expected asset loss incurred as a result of MaRV and BRV leakage. Because the interactions *between* individual engagements are considerably weaker than interactions *within* individual engagements, we have emphasized one-on-one engagements in this report. Their outcome is judged by the probability of kill achievable by an interceptor allocated to a given MaRV (or BRV). This probability is derived from a statement about the lethality of the interceptor and the miss distance achievable by the interceptor. We have intentionally left lethality discussions for future reports, and have concentrated on developing a method for predicting miss distances.

The algorithm we developed for predicting miss distances—called the MD Algorithm—uses geometric and kinematic information about the interceptor and the MaRV to produce a "best" estimate of miss distance from the most recent MaRV state update and from technological parameters and constraints. In predicting miss distances, MD accounts for information and response time delays between threat objects, sensors, and interceptors. It also contains a sensitivity framework with which the propagation of errors and uncertainties from data inputs to output decisions can be evaluated, but this capability has not yet been implemented in the software.

The MD algorithm was implemented and thoroughly tested, and several conclusions became apparent. For typical system time delays of 0.1 sec and for nominal MaRV and interceptor capabilities (see Chapter 4), the minimum miss distance achievable for "late" engagements (those occurring below 80,000 feet altitude) was approximately 600 feet. For "early" engagements MaRV maneuvering was limited and, if the MaRV was reachable by the interceptor, miss distances were less than 20 feet.

Another observation relates to the relative sensitivity of miss distance to interceptor velocity and maneuvering capabilities. It was found that the miss distance was considerably more sensitive to the turning capability of the vehicle than to vehicle speed. In fact, miss distance was relatively insensitive to MaRV speed. This suggests that current development efforts to improve vehicle maneuverability could be more effective than those to improve their speed.

Several things remain to be done. Foremost among these is the thorough testing of the battle manager as a whole. While its most important module—the MD Algorithm—was carefully tested as an individual unit, the battle manager has not yet been completely tested with that unit in place. The MD algorithm involves some heuristics, and several parameters must be set while it is operating as an integral part of the battle manager.

Such testing may also suggest ways to improve any of the algorithms discussed in this report. For instance, the straight-line MaRV trajectory prediction used in MD might be constrained or improved by using additional information about the destination of the MaRV. Further simulation may also suggest that the maneuvering test should be improved, but we believe that this is unlikely for the types of objects discussed in this report.

Acknowledgments

This report reflects several contributions by others. Foremost, the author would like to thank Michael Gorvad and Richard Pember for their perseverance, patience, and understanding in getting the battle managment and miss distance algorithms to execute on our SUN and VAX workstations, in spite of numerous minor—and some major—algorithmic aberrations for which the author is solely responsible. Also important were discussions held with Drs. L. Ng and S. Peglow, and A. Parziale, during which significant improvements to our work were suggested. Finally, we would like to thank F. McFarland for her text editing and production work, D. Gallant for her graphics, and W. Clements for his technical editing of this report.

References

[1] Weisenberger, David, Cindy Sims, and Tim Tolar, "Maneuvering Reentry Vehicle Attack/Defense Effectiveness Model—MaRVADEM," *Report No. AD-B088-468*, Sparta, Inc., Defense Technical Information Center, Huntsville, AL, October 1983.

[2] Jaquette, S., private communications, April 1988.

[3] Critchlow, Carl L. and Ronald C. Williams, "A Simulation Model for Analyzing Reentry Vehicle/Antiballistic Missile Engagements," *Report No. Ad-A115691*, Graduate Strategic and Tactical Sciences, USAF, March 1982.

[4] Dobbins, Harry M. and Michael J. Noviskey, "Trajectory Assessment and Threat Priorization," Technical Report AFAL-TR-79-1242, *Report No. AD-A088707*, Air Force Avionics Laboratory, Wright-Patterson AFB, Ohio, February 1980.

[5] Satterfield, Doyce E., "Resource Allocation and Scheduling in Ballistic Missile Defense Adaptive Control Systems," *Report No. AD-785-676*, Army Advanced Ballistic Missile Defense Agency, Huntsville, AL, 1974.

[6] Keearns, L. and S. Jaquette, "ERASE: An Overview," *Report No. ESD-TR-77-270*, Systems Control, Inc., Palo Alto, CA, October 1977.

[7] Burr, Stefan A., James E. Falk, and Alan F. Karr, "Integer Prim-Read Solutions to a Class of Target Defense Problems," *Report No. AD-A134191*, Institute for Defense Analyses, Program Analysis Division, September 1983.

[8] Corynen, G.C., "An Optimal Fire-Control Algorithm for Allocating *M* Beams to *N* targets," *Report UCRL-53850*, Lawrence Livermore National Laboratory, Livermore, CA, January 1988.

[9] Handler, Francis A., "Maneuverable Reentry Vehicle Trajectory Footprints: Calculation and Properties," *Report No. UCID-21286*, Lawrence Livermore National Laboratory, Livermore, CA, November 1987.

[10] Press, William H., B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes: The Art of Scientific Computing*, Cambridge University Press, 1986.

[11] Weast, Robert C. (Ed.) and M. J. Astle (Assoc. Ed.), *CRC Handbook of Chemistry and Physics*, CRC Press, Inc., 1981.

[12] Kelly, Michael, private communication, Lawrence Livermore National Laboratory, Livermore, CA, June 1987.

[13] Rajan, N. and M. D. Ardema, "Interception in Three Dimensions: An Energy Formulation," *J. Guidance*, Vol. 8, No. 1, pp. 23–30, Jan-Feb 1985.

[14] Nesline, F. William and Paul Zarchan, "Missile Guidance Design Tradeoffs for High-Altitude Air Defense," *J. Guidance*, Vol. 6, No. 3, pp. 207–212, May-June 1983.

[15] Chang, C. B., K. P. Dunn, and D. Willner, "Maneuvering Re-Entry Vehicle Engagement Miss Distance Achievable by Trilateration Radar Tracking," *Project Report No. RMP-100*, Massachusetts Institute of Technology, Lincoln Laboratory, Lexington, MA, September 16, 1976.

[16] Hudson, B. and M. Mintz, "Development of a Real-Time Global Decision Algorithm for Missile Evasion Phase I–A Look-Up Table Concept Development Study," *Technical Report No. AFWAL-TR-83-3054*, Flight Dynamics Laboratory, AF Wright Aeronautical Laboratories, Wright-Patterson AFB, OH, May 1983.

[17] Davidovitz, A. and J. Shinar, "Eccentric Two-Target Model for Qualitative Air Combat Game Analysis," *J. Guidance*, Vol. 8, No. 3, pp. 325–331, May-June 1985.

[18] Kim, Y. S., H. S. Cho, and Z. Bien, "A New Guidance Law for Homing Missiles," *J. Guidance*, Vol. 8, No. 3, pp. 402–404, May-June 1985.

[19] Fleming, Edward, "MaRV Engagement Analysis Study." Final Report, SAIC, Orlando, FL 32826, April 25, 1988.

[20] Gavel, Donald T., "Generating Optimally Evasive MaRV Trajectories," Lawrence Livermore National Laboratory (in progress).

APPENDIX A

Coordinate Systems and Transformations

A.1. Global (Reference) Coordinate System (R_e)

The global coordinate system is the earth center frame as shown in Figure A.1 [A1]. The earth center frame is a rectangular Cartesian frame whose origin is centered at earth center, the X-axis (X_e) is determined by the earth center and the point 0° latitude and 0° longitude, the Z-axis (Z_e) lies on the equatorial plane and at 90° from the x-axis. Finally, the Y-axis (Y_e) connects the earth center to the north pole.

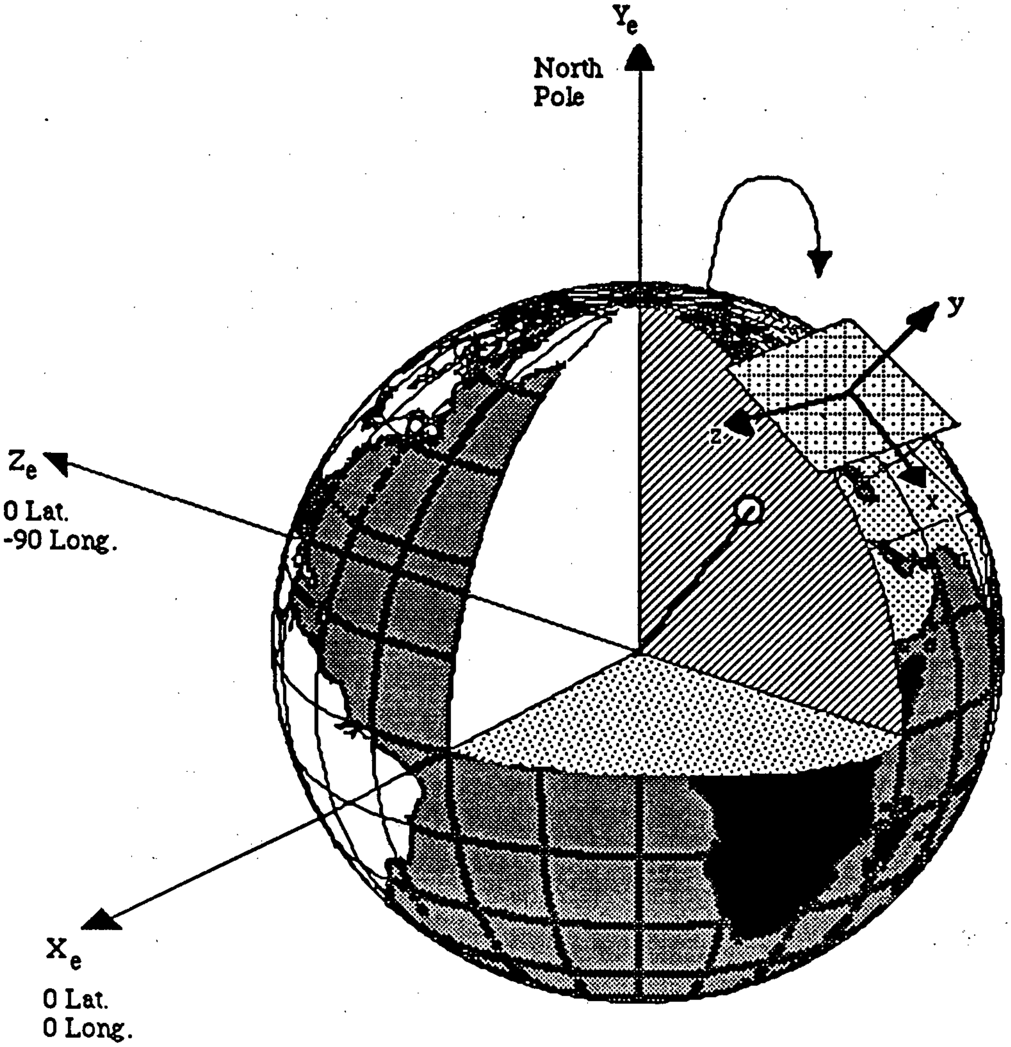


Figure A.1. Specifications of global coordinate system and local level frame.

A.2. Local (Reference) Coordinate System (R-System)

As seen in Figure A.1, the x and z axes of this system lie on a plane tangent to the earth at a point that is "arbitrarily" defined as the origin of the local coordinate system. The x-axis in this R-system always points south and aligns with the longitudinal meridian. The y-axis points up, and the z-axis points west.

The origin is usually chosen to be the centroid of the asset locations.

A.3. MaRV Coordinate System (M-System)

The M-system is centered at the MaRV position vector $X_M(\Delta t_M)$, expressed in the R-system.

If e_y is a unit vector along the y-axis (of the R-system), and $e_{V_M^0}$ is a unit vector along the MaRV velocity vector V_M^0, then the unit vector triple $(e_{x_M}, e_{y_M}, e_{z_M})$ along the MaRV axis triple (x_M, y_M, z_M) is defined as follows:

$$e_{x_M} = \frac{V_M^0}{|V_M^0|} = e_{V_M^0} \quad , \tag{A1}$$

$$e_{y_M}(t_w) = e_{z_M}(t_w) \times e_{x_M} \quad , \tag{A2}$$

$$e_{z_M}(t_w) = \begin{cases} \dfrac{V_M^0 \times e_x}{|V_M^0 \times e_x|}: \ (V_M^0 \times e_x) \cdot e_y > 0 \bigwedge V_M^0 \times \Delta X_I(t_w) = 0 \\[2ex] -\dfrac{V_M^0 \times e_x}{|V_M^0 \times e_x|}: \ (V_M^0 \times e_x) \cdot e_y \leq 0 \bigwedge V_M^0 \times \Delta X_I(t_w) = 0 \\[2ex] \dfrac{\Delta X_I(t_w) \times V_M^0}{|\Delta X_I(t_w) \times V_M^0|}: \ \left(\Delta X_I(t_w) \times V_M^0\right) \cdot e_y > 0 \\[2ex] -\dfrac{\Delta X_I(t_w) \times V_M^0}{|\Delta X_I(t_w) \times V_M^0|}: \ \left(\Delta X_I(t_w) \times V_M^0\right) \cdot e_y \leq 0 \end{cases} \quad , \tag{A3}$$

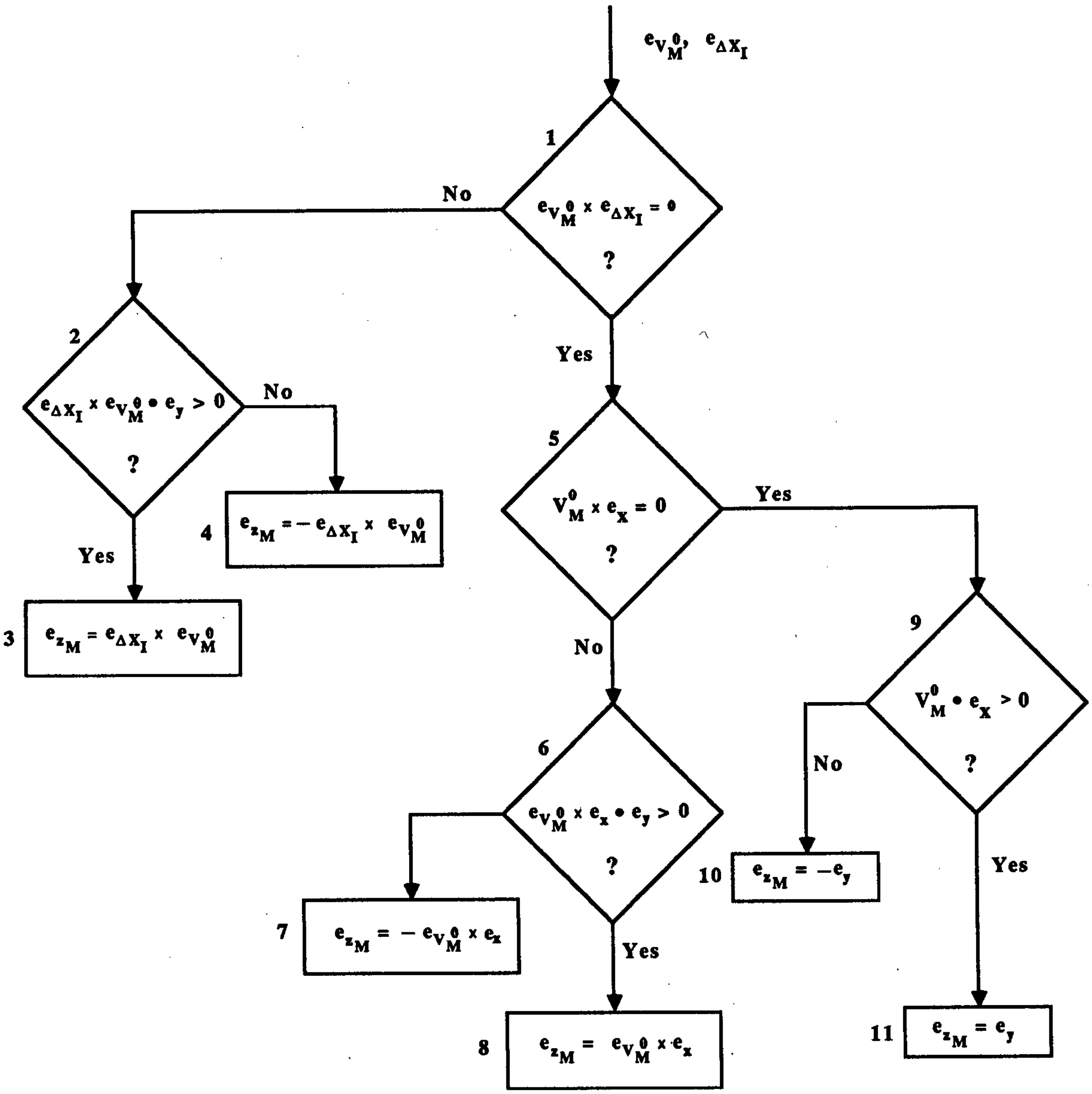
where $e_{\Delta X_I}$ is a unit vector along the interceptor launch direction vector ΔX_I. Note that for each new value of t_{wait} we get a new coordinate system $(x_M, y_M, z_M)(t_{\text{wait}})$.

The purpose for defining the M-system this way is to facilitate the analysis and computation of MaRV and interceptor maneuvers. The nominal plane in which these maneuvers take place (the nominal "action plane") is the $(e_{V_M^0}, e_{\Delta X_I})$-plane. When maneuvers are not in that plane, they are transformed into that plane by the rotation R discussed below.

In the definition of e_{z_M}, a test is needed to avoid conditions where zero vectors would be chosen as unit coordinate vectors, as would be the case when $e_{\Delta X_I}$ and $e_{V_M^0}$ are collinear, *i.e.*, $e_{V_M^0} \times e_{\Delta X_I} = 0$.

Another test is needed to assure that e_{z_M} is pointing in the right direction, and this requires calculating the dot products with e_y, as shown.

To minimize computation time in determining the M-coordinates as the engagement unfolds, we use the following flowchart (Figure A.2).



$e_{V_M^0}$, $e_{\Delta X_I}$
1
$e_{V_M^0} \times e_{\Delta X_I} = 0$?
No
Yes
2
$e_{\Delta X_I} \times e_{V_M^0} \bullet e_y > 0$?
No
Yes
3
$e_{z_M} = e_{\Delta X_I} \times e_{V_M^0}$
4
$e_{z_M} = - e_{\Delta X_I} \times e_{V_M^0}$
5
$V_M^0 \times e_x = 0$?
Yes
No
6
$e_{V_M^0} \times e_x \bullet e_y > 0$?
Yes
7
$e_{z_M} = - e_{V_M^0} \times e_x$
8
$e_{z_M} = e_{V_M^0} \times e_x$
9
$V_M^0 \bullet e_x > 0$?
No
Yes
10
$e_{z_M} = -e_y$
11
$e_{z_M} = e_y$

Figure A.2. Determining the MaRV coordinate system.

A.4. Coordinate Transformations

Transformations are needed in the miss distance calculations because vehicle height (altitude) in the R-system is an important atmospheric parameter. Altitude is simply the projection V_{R_y} of the position vector V_R onto the y-coordinate and, because the curvature radius of the MaRV depends upon its altitude, which is expressed in the R-system, we need to transform M-points into R-points and vice-versa. These transformations require both a translation and a rotation. Consider Figure A.3 where a transformation from M to R is illustrated.

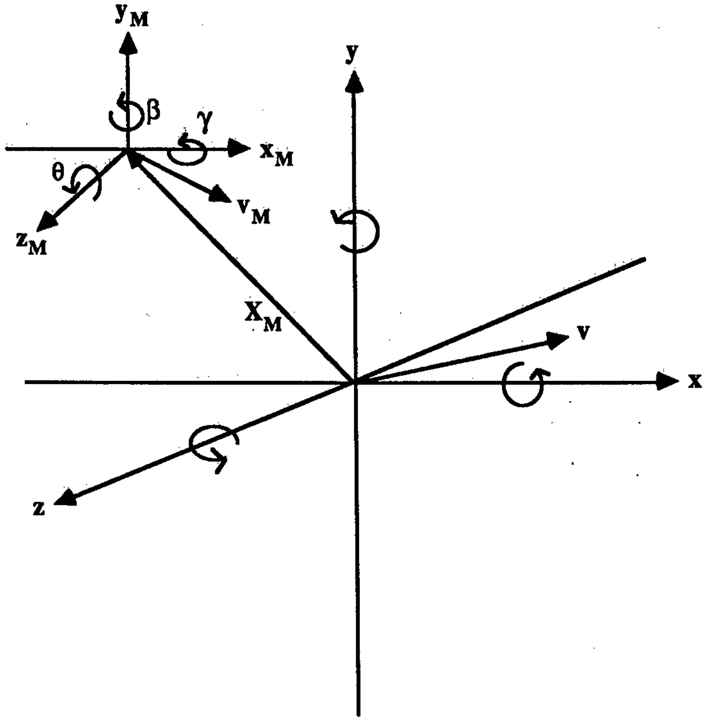


Figure A.3. Transforming a vector V_M in the M-coordinate system to a vector V in the R-system.

Rotations are done in three steps [A2].

1. Rotate V_M about the z_M-axis by the (elevation) angle θ between x_M and the x–z plane ($\sin(\theta) = -e_{V_M} \cdot e_y$), yielding a rotation matrix

$$R_{z_M}(\theta) = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad . \tag{A4}$$

2. Rotate the rotated vector $R_{z_M}(\theta)\cdot V_M$ about the y_M axis by an angle β between x_M and the x–y plane, yielding a rotation matrix

$$R_{y_M}(\beta) = \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix} \quad . \tag{A5}$$

3. Rotate the rotated vector $R_{y_M}(\beta)\cdot R_{z_M}(\beta)\cdot V_M$ about the x_M axis by an angle γ between z_M and the x–y plane, yielding a rotation matrix

$$R_{z_M}(\gamma) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\gamma & \sin\gamma \\ 0 & -\sin\gamma & \cos\gamma \end{bmatrix} \quad . \tag{A6}$$

The composite rotation of V_M thus yields a vector

$$V_R = R_{z_M}(\gamma)\cdot R_{y_M}(\beta)\cdot R_{x_M}(\theta)V_M \quad , \tag{A7}$$

in the R-system.

Since the origins of the M-system and the R-system are different, we also need to translate V_M. To be consistent with the meaning of the rotation angles, this must be done first, before rotating V_M. The overall transformation is thus

$$V_R = T(V_M) = R_{z_M}(\gamma)\cdot R_{y_M}(\beta)\cdot R_{x_M}(\theta)[V_M - X_M] \quad , \tag{A8}$$

where X_M is the position vector of the M-system in the R-system. Thus

$$V_{R_y} = \mathrm{Proj}_y \left[R_{z_M}(\gamma)\cdot R_{y_M}(\beta)\cdot R_{x_M}(\theta)\left[V_M - X_M\right]\right] \quad . \tag{A9}$$

References

[A1] Ng, L. C., "Coordinate System Specification," Internal Memorandum Notes, Lawrence Livermore National Laboratory, Livermore, CA, July 29, 1987.

[A2] Noble, Ben and J. W. Daniel, *Applied Linear Algebra,* Prentice-Hall Inc., 1977.

APPENDIX B

Simplified Equations of Motion for Interceptor Type I (High Altitude)

B.1. Mass (Figure B.1)

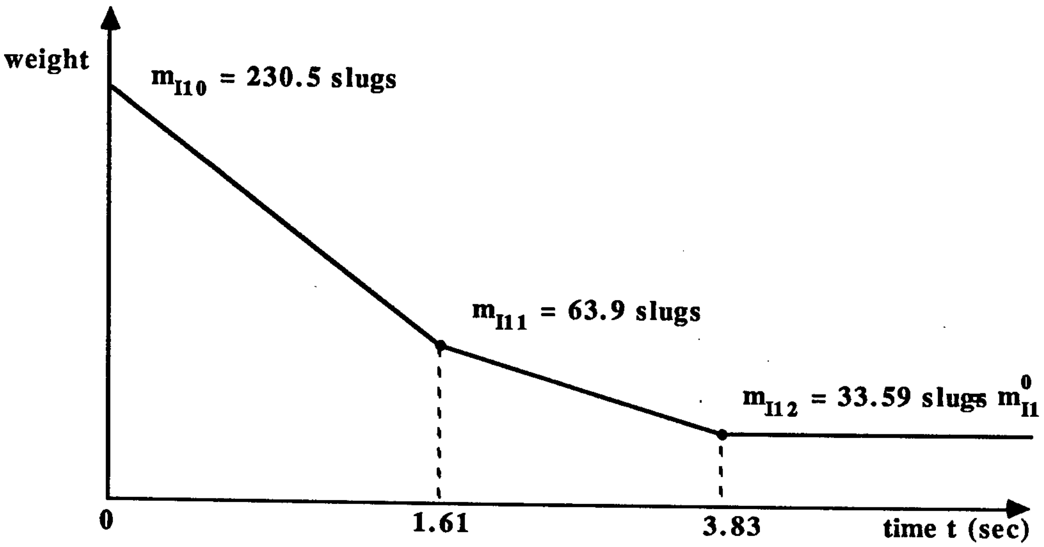


Figure B.1. Weight profile of a type I interceptor.

$$m_{I1}(t) = X_1(t) * m_{I11}(t) + Y_1(t) * m_{I12}(t) + Z_1(t) * m_{I13}(t) \quad , \qquad \text{(B1)}$$

$$X_1(t) = \frac{1 + \text{sign}(1.61 - t)}{2} \quad , \qquad \text{(B2)}$$

$$Y_1(t) = \left(\frac{1 + \text{sign}(3.83 - t)}{2}\right)\left(\frac{1 + \text{sign}(t - 1.61)}{2}\right) \quad , \qquad \text{(B3)}$$

$$Z_1(t) = \left(\frac{1 + \text{sign}(t - 3.83)}{2}\right) \quad , \qquad \text{(B4)}$$

$$m_{I11}(t) = 230.5(1 - 0.449t) + (m^0_{I1} - 33.59) \quad , \quad 0 \leq t \leq 1.61 \quad , \qquad \text{(B5)}$$

$$m_{I12}(t) = 85.91(1 - 0.159t) + (m^0_{I1} - 33.59) \quad , \quad 1.61 \leq t \leq 3.83 \quad , \qquad \text{(B6)}$$

$$m_{I13}(t) = m^0_{I1} \quad , \quad 3.83 \leq t \quad . \qquad \text{(B7)}$$

B.2. Thrust (Figure B.2)

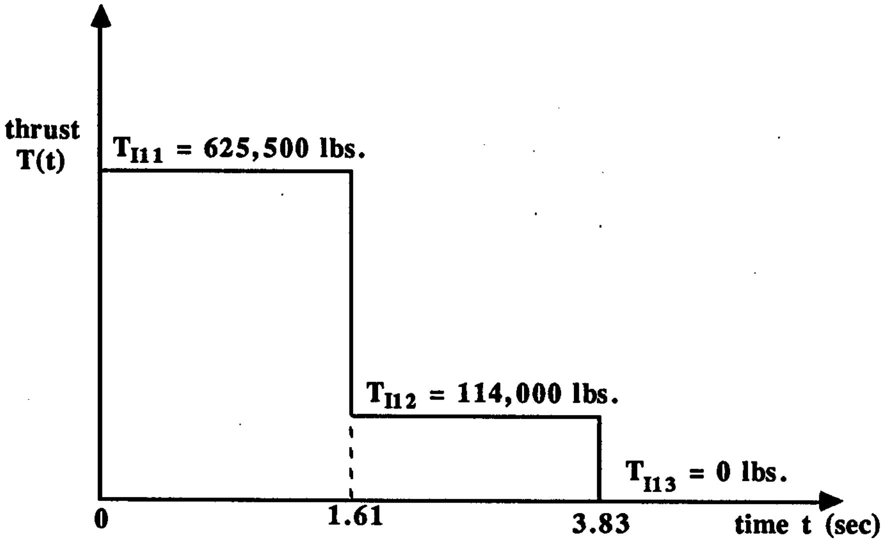


Figure B.2. Thrust profile of a type I interceptor.

$$T_{I1}(t) = X_1(t) * T_{I11}(t) + Y_1(t) * T_{I12}(t) + Z_1(t) * T_{I13}(t) \quad , \qquad \text{(B8)}$$

$$T_{I21}(t) = 625,500 \quad , \quad 0 \le t \le 1.61 \quad , \qquad \text{(B9)}$$

$$T_{I12}(t) = 114,000 \quad , \quad 1.61 \le t \le 3.83 \quad , \qquad \text{(B10)}$$

$$T_{I13}(t) = 0 \quad , \quad 3.83 \le t \quad . \qquad \text{(B11)}$$

B.3. Acceleration Due to Thrust

$$a_{I1}(t) = X_1(t) * a_{I11}(t) + Y_1(t) * a_{I12}(t) + Z_1(t) * a_{I13}(t) \quad , \qquad \text{(B12)}$$

$$a_{I11}(t) = \frac{625,500}{230.5(1 - 0.449t) + (m_{I1}^0 - 33.59)} \quad , \quad 0 \le t \le 1.61 \quad , \qquad \text{(B13)}$$

$$a_{I12}(t) = \frac{114,000}{85.91(1 - 0.159t) + (m_{I1}^0 - 33.59)} \quad , \quad 1.61 \le t \le 3.83 \quad , \qquad \text{(B14)}$$

$$a_{I13}(t) = 0 \quad , \quad 3.83 \le t \quad . \qquad \text{(B15)}$$

B.4. Speed [B1,B2]

$$V_{I1}(t) = X_1(t) * V_{I11}(t) + Y_1(t) * V_{I12}(t) + Z_1(t) * V_{I13}(t) \quad , \tag{B16}$$

$$\text{Fact:} \quad \int \frac{k}{a+bx} dx = \frac{k}{b} \ln(a+bx) \quad , \tag{B17}$$

$$V_{I11}(t)\text{:} \quad a = m^0_{I1} + 196.9 \quad , \quad b = -103.5 \quad , \quad \frac{k}{b} = -6043 \quad . \tag{B18}$$

$$\begin{aligned} V_{I11}(t) &= -6043 \ln(m^0_{I1} + 196.9 - 103.5\tau)\Big|_0^t \\ &= -6043[\ln(m^0_{I1} + 196.9 - 103.5t) - \ln(m^0_{I1} + 196.9)] \quad , \quad 0 \le t \le 1.61 \quad . \end{aligned} \tag{B19}$$

$$V_{I12}(t)\text{:} \quad a = m^0_{I1} + 52.32 \quad , \quad b = -13.66 \quad , \quad \frac{k}{b} = -8346 \quad . \tag{B20}$$

$$\begin{aligned} V_{I12}(t) &= -8346 \ln(m^0_{I1} + 52.32 - 13.66\tau)\Big|_{\tau=1.61}^t + V_{I11}(1.61) \\ &= -6043[\ln(m^0_{I1} + 30.26) - \ln(m^0_{I1} + 196.9)] + 8346 \ln(m^0_{I1} + 30.33) \\ &\quad - 8346 \ln(m^0_{I1} + 52.32 - 13.66t) \quad , \quad 1.61 \le t \le 3.83 \quad . \end{aligned} \tag{B21}$$

$$\begin{aligned} V_{I12}(t) &= 2303 \ln(m^0_{I1} + 30.30) + 6043 \ln(m^0_{I1} + 196.9) \\ &\quad - 8346 \ln(m^0_{I1} + 52.32 - 13.66t) \quad , \quad 1.61 \le t \le 3.83 \quad . \end{aligned} \tag{B22}$$

$$V_{I13}(t) = \begin{cases} V^0_{I13}\left[1 - \left(\frac{t-3.83)}{k_{DI1}}\right)\right] \quad , & 3.83 \le t \le 3.83 + k_{DI_1} \\ 0 \ , & 3.83 + k_{DI_1} \le t \end{cases} \quad , \tag{B23}$$

where

$$V^0_{I13} = V_{I12}(3.83) = 2303 \ln(m^0_{i1} + 30.30) + 6043 \ln(m^0_{I1} + 196.9) - 8346 \ln(m^0_{I1}) \quad .$$

B.5. Position [B1,B2]

$$X_{I1}(t) = X_1(t) * X_{I11}(t) + Y_1(t) * X_{I12}(t) + Z_1(t) * X_{I13}(t) \quad , \tag{B24}$$

$$\text{Fact:} \quad \int \ln(ax+b) = \left(\frac{ax+b}{a}\right) \ln(ax+b) - x \quad , \tag{B25}$$

$$X_{I11}(t)\text{:} \quad a = -103.5 \quad , \quad b = m^0_{I1} + 196.9 \quad . \tag{B26}$$

$$X_{I11}(t) = 6043\left\{\ln(m_{I1}^0 + 196.9)t + \frac{(m_{I1}^0 + 196.9 - 103.5\tau)}{103.5}\ln(m_{I1}^0 + 196.9 - 103.5\tau) + \tau\Big|_0^t\right\}$$

$$= 6043\Bigg\{\left(\ln\left(m_{I1}^0 + 196.9\right) + 1\right)t + \frac{(m_{I1}^0 + 196.9 - 103.5t)}{103.5}\ln(m_{I1}^0 + 196.9 - 103.5t)$$

$$- \left(\frac{m_{I1}^0 + 196.9}{103.5}\right)\ln(m_{I1}^0 + 196.9)\Bigg\}$$

$$= 6043\Bigg\{\ln(m_{I1}^0 + 196.9) + 1]t + \left(\frac{m_{I1}^0}{103.5} + 1.90 - t\right)\ln(m_{I1}^0 + 196.9 - 103.5t)$$

$$- \left(\frac{m_{I1}^0 + 196.9}{103.5}\right)\ln(m_{I1}^0 + 196.9)\Bigg\} \quad , \quad 0 \le t \le 1.61 \quad . \qquad \text{(B27)}$$

$$X_{I12}(t)\text{:}\ \ a = -13.66 \quad , \quad b = m_{I1}^0 + 52.32 \quad . \qquad \text{(B28)}$$

$$X_{I12}(t) = X_{I11}(1.61) + \int_{\tau=1.61}^{t} V_{I12}(\tau)d\tau$$

$$= X_{I11}(1.61) + [2303\ln(m_{I1}^0 + 30.30) + 6043\ln(m_{I1}^0 + 196.9)]\tau\Big|_{1.61}^{t}$$

$$+ 8346\left[\left(\frac{m_{I1}^0 + 52.32 - 13.66\tau}{+13.66}\right)\ln(m_{I1}^0 + 52.32 - 13.66\tau) + \tau\right]_{1.61}^{t}$$

$$= X_{I11}(1.61) + [2303\ln(m_{I1}^0 + 30.30) + 6043\ln(m_{I1}^0 + 196.9) + 8346]t$$

$$- [3708\ln(m_{I1}^0 + 30.30) + 9729\ln(m_{I1}^0 + 13,437]$$

$$+ 8346\Bigg[\left(\frac{m_{I1}^0}{13.66} + 3.83 - t\right)\ln(m_{I1}^0 + 52.32 - 13.66t)$$

$$- \left(\frac{m_{I1}^0}{13.66} + 2.22\right)\ln(m_{I1}^0 + 30.30)\Bigg] \quad , \qquad \text{(B29)}$$

$$X_{T12}(t) = X_{I11}(1.61) + [2303\ln(m_{I1}^0 + 30.30) + 6043\ln(m_{I1}^0 + 196.9) + 8346]t$$

$$+ 8346\left(\frac{m_{I1}^0}{13.66} + 3.83 - t\right)\ln(m_{I1}^0 + 52.32 - 13.66t)$$

$$- (611.0\, m_{I1}^0 + 22,238)\ln(m_{I1}^0 + 30.30)$$

$$+ 9729\ln(m_{I1}^0 + 196.9) + 13,437 \quad , \quad 1.61 \le t \le 3.83 \quad . \qquad \text{(B30)}$$

$X_{I13}(t)$:

$$X_{I13}(t) = X_{I12}(3.83) + \int_{\tau=3.83}^{t} V_{I13}(\tau)d\tau = X_{I12}(3.83) + V_{I13}^{0}\left[\left(1+\frac{3.83}{k_{DI1}}\right)\tau - \frac{\tau^2}{2k_{DI1}}\right]_{3.83}^{t}$$

$$= X_{I12}(3.83) + V_{I13}^{0}\left[\left(1+\frac{3.83}{k_{DI1}}\right)t - \frac{t^2}{2k_{DI1}} - 3.83\left(1+\frac{3.83}{2k_{DI1}}\right)\right] \quad ,$$

$$3.83 \le t \le 3.83 + k_{DI1} \quad ,$$

$$X_{I13}(t) = X_{I13}(3.83 + k_{DI1}) \quad , \quad 3.83 + k_{DI1} \le t \quad . \tag{B31}$$

References

[B1] Gradshteyn, I. S. and I. M. Ryzhik, *Table of Integrals, Series, and Products,* Academic Press, Inc., Orlando, FL, 1980.

[B2] Abramowitz, Milton and Irene A. Stegun (Eds.), *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables,* National Bureau of Standards Applied Mathematics Series 55, December 1972.

APPENDIX C

Simplified Equations of Motion for Interceptor Type II (Low Altitude)

C.1. Weight/Mass (Figure C.1)

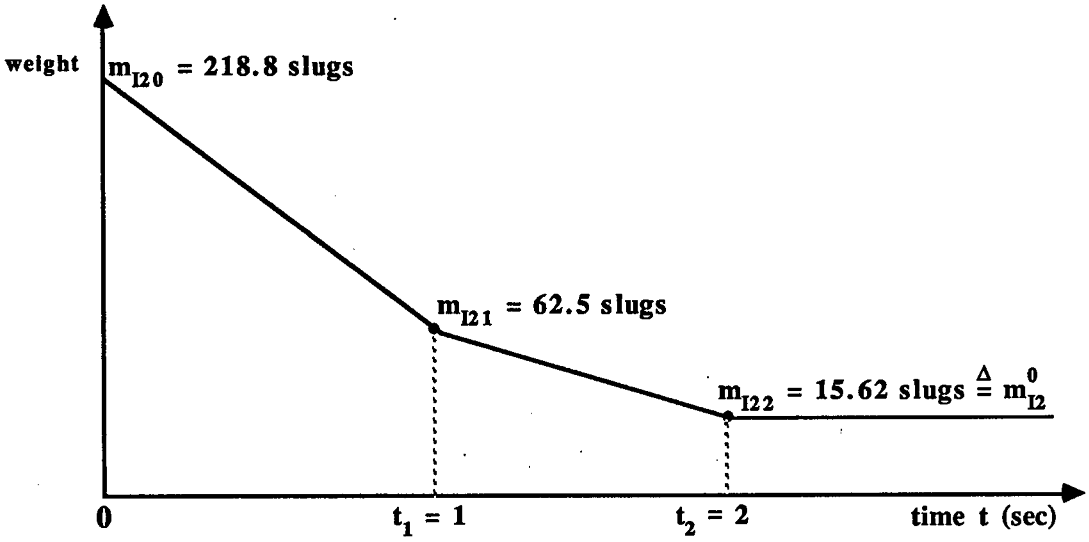


Figure C.1. Weight profile of a short-range type II interceptor.

$$m_{I2}(t) = X_2^{(t)} * m_{I11}(t) + Y_2^{(t)} * m_{I22}(t) + Z_2^{(t)} * m_{I23}(t) \quad , \tag{C1}$$

$$X_2(t) = \frac{1 + \text{sign}(1-t)}{2} \quad , \tag{C2}$$

$$Y_2(t) = \left(\frac{1 + \text{sign}\,(2-t)}{2}\right)\left(\frac{1 + \text{sign}\,(t-1)}{2}\right) \quad , \tag{C3}$$

$$Z_2(t) = \left(\frac{1 + \text{sign}\,(t-2)}{2}\right) \quad , \tag{C4}$$

$$m_{I21}(t) = 218.8(1 - 0.714t) + m^0_{I2} - 15.62 \quad , \quad 0 \le t \le 1 \quad , \tag{C5}$$

$$m_{I22}(t) = 109.4(1 - 0.429t) + m^0_{I2} - 15.62 \quad , \quad 1 \le t \le 2 \quad , \tag{C6}$$

$$m_{I23}(t) = m^0_{I2} \quad , \quad 2 \le t \quad . \tag{C7}$$

C.2. Thrust (Figure C.2)

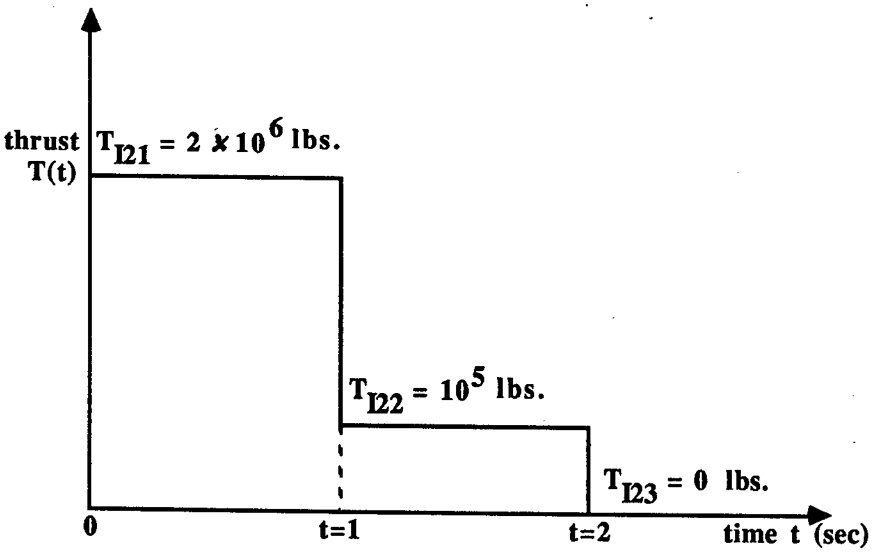


Figure C.2. Thrust profile of a short-range type II interceptor.

$$T_{I2}(t) = X_2(t) * T_{I21}(t) + Y_2(t) * T_{I22}(t) + Z_2(t) * T_{I23}(t) \quad , \tag{C8}$$

$$T_{I21}(t) = 2 \times 10^6 \quad , \quad 0 \leq t \leq 1 \quad , \tag{C9}$$

$$T_{I22}(t) = 10^5 \quad , \quad 0 \leq t \leq 2 \quad , \tag{C10}$$

$$T_{I23}(t) = 0 \quad , \quad 2 \leq t \leq \infty \quad . \tag{C11}$$

C.3. Acceleration Due to Thrust

$$a_{I2}(t) = X_2(t) * a_{I21}(t) + Y_2(t) * a_{I22}(t) + Z_2(t) * a_{I23}(t) \quad , \tag{C12}$$

$$a_{I21}(t) = \frac{2 \times 10^6}{218.8(1 - 0.714t) + m_{I2}^0 - 15.62} \quad , \quad 0 \leq t \leq 1 \quad , \tag{C13}$$

$$a_{I22}(t) = \frac{10^5}{109.4(1 - 0.429t) + m_{I2}^0 - 15.62} \quad , \quad 1 \leq t \leq 2 \quad , \tag{C14}$$

$$a_{I23}(t) = 0 \quad , \quad 2 \leq t \quad . \tag{C15}$$

C.4. Velocity (Speed) [B1,B2]

$$V_{I2}(t) = X_2(t) * V_{I21}(t) + Y_2(t) * V_{I22}(t) + Z_2(t) * V_{I23}(t) \quad . \tag{C16}$$

$$\text{Fact:} \quad \int \frac{k dx}{a + bx} = \frac{k}{b} \ln(a + bx) \quad , \tag{C17}$$

$$V_{I21}(t)\text{:} \quad a = 203.18 + m_{I2}^0 \quad , \quad b = -156.2 \quad , \quad \frac{k}{b} = -12,803 \quad . \tag{C18}$$

$$\begin{aligned} V_{I21}(t) &= -12,803 \ln(203.18 + m_{I2}^0 - 156.2\tau)\Big|_0^t \\ &= -12,803[\ln(203.18 + m_{I2}^0 - 156.2t) - \ln(203.18 + m_{I2}^0)] \ , \ 0 \leq t \leq 1 \ . \end{aligned} \tag{C19}$$

$$V_{I22}(t)\text{:} \quad a = 93.78 + m_{I2}^0 \quad , \quad b = -46.98 \quad , \quad \frac{k}{b} = -2131 \quad , \tag{C20}$$

$$\begin{aligned} V_{I22}(t) = {} & 12,803[\ln(203.18 + m_{I2}^0) - \ln(46.98 + m_{I2}^0)] \\ & - 2131[\ln(93.78 + m_{I2}^0 - 46.98) - \ln(46.98 + m_{I2}^0)] \quad , \quad 1 \leq t \leq 2 \quad . \end{aligned} \tag{C21}$$

$$\begin{aligned} V_{I23}(t) &= \left\{ \begin{array}{l} 12,803[\ln(203.18 + m_{I2}^0) - \ln(46.98 + m_{I2}^0)] \\ -2131[\ln(m_{I2}^0) - \ln(46.98 + m_{I2}^0)] \end{array} \right\} \left\{ \left(1 - \frac{(t-2)}{k_{DI2}} \right) \right\} \\ &= V_{I23}^0 \left[\left(1 - \left(\frac{t-2}{k_{DI2}} \right) \right) \right] \quad , \end{aligned} \tag{C22}$$

$$V_{I23}^0 = 12,803 \ln(203.18 + m_{I2}^0) - 10,672 \ln(46.98 + m_{I2}^0) - 2131 \ln(m_{I2}^0) \ , \ 2 \leq t \quad . \tag{C23}$$

C.5. Position [B1,B2]

$$X_{I2}(t) = X_2(t) * X_{I21}(t) + Y_2(t) * X_{I22}(t) + Z_2(t) * X_{I23}(t) \quad , \tag{C24}$$

$$\text{Fact:} \quad \int k \ln(ax + b) dx = k \left[\left(\frac{ax + b}{a} \right) \ln(ax + b) - x \right] \quad , \tag{C25}$$

$$X_{I21}(t)\text{:} \quad a = -156.2 \quad , \quad b = 203.18 + m_{I2}^0 \quad , \quad k = -12,803 \quad . \tag{C26}$$

$$\begin{aligned} X_{I21}(t) = 12,803 \Bigg\{ & \ln(203.18 + m_{I2}^0)\tau \Big|_0^t \\ & + \left[\left(\frac{203.18 + m_{I2}^0 - 156.2\tau}{+156.2} \right) \ln\left(203.18 + m_{I2}^0 - 156.2\tau\right) + \tau \right] \Big|_0^t \Bigg\} \end{aligned}$$

$$= 12,803\left\{\ln(203.18 + m^0_{I2})t + t + \left(1.3 + \frac{m^0_{I2}}{156.2} - t\right)\ln(203.18 + m^0_{I2} - 156.2t)\right.$$

$$\left. - \left(1.3 + \frac{m^0_{I2}}{156.2}\right)\ln(203.18 + m^0_{I2})\right\}$$

$$= 12,803\left\{\left(\ln\left(203.18 + m^0_{I2}\right) + 1\right)t - \left(1.3 + \frac{m^0_{I2}}{156.2}\right)\ln(203.18 + m^0_{I2})\right.$$

$$\left. + \left(1.3 + \frac{m^0_{I2}}{156.2} - t\right)\ln(203.18 + m^0_{I2} - 156.2t)\right\} \ , \ 0 \le t \le 1 \quad . \qquad \text{(C27)}$$

$$X_{I22}(t)\text{:} \ \ a = -46.9 \quad , \quad b = 93.78 + m^0_{I2} \quad , \quad k = -2131 \quad , \qquad \text{(C28)}$$

$$X_{I22}(t) = X_{I21}(1) + \int_{\tau=1}^{t} V_{I22}(\tau)d\tau$$

$$= X_{I21}(1) + \tau[12,803\ln(203.18 + m^0_{I2}) - 10,672\ln(46.98 + m^0_{I2})]\Big|_1^t$$

$$- 2131\left[\left(\frac{46.9\tau - (93.78 + m^0_{I2})}{46.9}\right)\ln(93.78 + m^0_{I2} - 46.9\tau) - \tau\right]_1^t \qquad \text{(C29)}$$

$$= X_{I21}(1) + \left(12,803\ln\left(203.18 + m^0_{I2}\right) - 10,672\ln\left(46.98 + m^0_{I2}\right) + 2131\right)t$$

$$- 12,803\ln(203.18 + m^0_{I2}) - 2131 + (8541 - 45.44\,m^0_{I2})\ln(46.98 + m^0_{I2})$$

$$+ (4261 + 45.44\,m^0_{I2} - 2131t)\ln(93.78 + m^0_{I2} - 46.9t) \ , \ 1 \le t \le 2 \ . \quad \text{(C30)}$$

$X_{I23}(t)$:

$$X_{I23}(t) = X_{I22}(2) + \int_{\tau=2}^{t} V_{I23}(\tau)d\tau = X_{I22}(2) + V^0_{I23}\left[\left(1 + \frac{2}{k_{DI2}}\right)\tau - \frac{\tau^2}{2k_{DI2}}\right]_2^t$$

$$= X_{I22}(2) - V^0_{I23}\left(\frac{2}{k_{DI2}} + 2 - \left(1 + \frac{2}{k_{DI2}}\right)t + \frac{t^2}{2k_{DI2}}\right) \ ,$$

$$2 \le t \le 2 + k_{DI2} \quad . \qquad \text{(C31)}$$

APPENDIX D

Some Derivatives for Using the Newton-Raphson Method in Solving $\mathbf{f(x) = 0}$.

In using the root solving routine RTSAFE [10], a significant speed-up is obtained if the derivatives of $f(x)$ are provided. In this appendix, we present these derivatives for the initialization problem.

"Solve:

$$F^0(t_{\text{wait}}, \Delta t_I) = d^0_{MI}(t_{\text{wait}}, \Delta t_I) - \Delta R(t_{\text{wait}}, \Delta t_I) = 0$$
$$\text{for } \Delta t_I,\ t_{\text{wait}} \text{ fixed."} \tag{D1}$$

We need the derivative $(F^0)'(t_{\text{wait}}, \Delta t_I)$ of $F^0(t_{\text{wait}}, \Delta t_I)$.

D.1.

$$F^0(t_{\text{wait}}, \Delta t_I) = d^0_{MI}(t_{\text{wait}}, \Delta t_I) - \Delta R(t_{\text{wait}}, \Delta t_I) \quad , \tag{D2}$$

$$(F^0)'(t_{\text{wait}}, \Delta t_I) = (d^0_{MI})'(t_{\text{wait}}, \Delta t_I) - \Delta R'(t_{\text{wait}}, \Delta t_I) \quad . \tag{D3}$$

D.2.

$$d^0_{MI}(t_{\text{wait}}, \Delta t_I) = 2[r_M(t_{\text{wait}}, \Delta t_I) r_I(t_{\text{wait}}, \Delta t_I)]^{1/2} \cos(\theta^0_c) + r_I(t_{\text{wait}}, \Delta t_I)(\sin\theta^0_c) \quad , \tag{D4}$$

where θ^0_c is the collision angle between the MaRV(M) and the interceptor (I).

Define

$$\Delta t_M \overset{\triangle}{=} t_{\text{wait}} + \Delta t_I \quad . \tag{D5}$$

Then

$$r_M(t_{\text{wait}}, \Delta t_I) = r_M(t_{\text{wait}} + \Delta t_I) = r_M(\Delta t_M) \quad ,$$

and

$$r_I(t_{\text{wait}}, \Delta t_I) = r_I(\Delta t_I) \quad . \tag{D6}$$

Now

$$(d^0_{MI})'(t_{\text{wait}}, \Delta t_I) = \frac{\partial d^0_{MI}}{\partial r_M} \frac{\partial r_M}{\partial \Delta t_I} + \frac{\partial d^0_{MI}}{\partial r_I} \frac{\partial r_I}{\partial \Delta t_I} \quad . \tag{D7}$$

D.3.

$$\frac{\partial d^0_{MI}}{\partial r_M} = (\cos\theta^0_c) \left(\frac{r_I(\Delta t_I)}{r_M(\Delta t_M)} \right)^{1/2} \quad , \tag{D8}$$

$$\frac{\partial d^0_{MI}}{\partial r_I} = (\cos\theta^0_c)\left(\frac{r_M(\Delta t_M)}{r_I(\Delta t_I)}\right)^{1/2} + \sin\,\theta^0_c \quad . \tag{D9}$$

D.4.

$$r_M(t_{\text{wait}},\, \Delta t_I) = r_M(\Delta t_I + t_{\text{wait}}) \equiv r_M(\Delta t_M) \quad , \tag{D10}$$

$$r_M(\Delta t_M) = \frac{2m_M\ \exp\left\{\dfrac{X_M(\Delta t_M)\bullet e_y}{h_0} - 1\right\}}{\rho_0 C_L(\alpha,\ \text{Mach})_M S_M} \quad , \tag{D11}$$

$$\frac{\partial r_M(\Delta t_M)}{\partial(\Delta t_I)} = \left(\frac{2m_M\ \exp\left\{\dfrac{X_M(\Delta t_M)\bullet e_y}{h_0} - 1\right\}}{\rho_0 C_L(\alpha,\ \text{Mach})_M S_M}\right)\left(\frac{\partial X_M(\Delta t_M)\bullet e_y}{h_0 \partial(\Delta t_I)}\right) \quad , \tag{D12}$$

$$\frac{\partial X_M(\Delta t_M)}{\partial(\Delta t_I)} = |V_M(\Delta t_M)| e_{V_M} \quad , \tag{D13}$$

$$|V_M(\Delta t_M)| = \begin{cases} |V^0_M|\ , & 0 \le \Delta t_M \le \tau_0 \\ |V^0_M|\left[1 - \frac{(\Delta t_M - \tau_0)}{k_{DM}}\right]\ , & 0 \le \tau_0 \le \Delta t_M \\ |V^0_M|\left[1 - \frac{\Delta t_M}{k_{DM}}\right]\ , & \tau_0 < 0 \le \Delta t_M \end{cases} \quad , \tag{D14}$$

where

$$\tau_0 = \frac{1}{V^0_M \bullet e_y}(H_0 - X^0_M \bullet e_y) \quad , \tag{D15}$$

$$\sin\theta^0_e = e_{X^0_M}\bullet e_y \quad , \quad e_{X^0_M} \stackrel{\triangle}{=} \frac{X^0_M}{|X^0_M|} \quad , \tag{D16}$$

$$k_{DM} = \frac{-2V^0_M \min\{H_0,\, X^0_M \bullet e_y\}}{e_{V^0_M}\bullet e_y\left(|V^0_M| - |V^F_M|\right)\left(|V^0_M| + |V^F_M|\right)} \quad , \tag{D17}$$

V^0_M is the speed of M at last observation, and V^F_M is the final (impact) velocity of M.

If $V^0_M \bullet e_y \ge -10^{-3}$, set $k_{DM} = 10^4$.

D.5.

$$r_I(\Delta t_I) = \frac{2m_I(\Delta t_I)\ \exp\left\{\dfrac{X_I(\Delta t_I)\bullet e_y}{h_0} - 1\right\}}{\rho_0 C_L(\alpha,\ \text{Mach})_I S_I} \quad , \tag{D18}$$

$$\frac{\partial r_I}{\partial \Delta t_I} = \frac{2\exp\left\{\dfrac{X_I(\Delta t_I)\cdot e_y}{h_0} - 1\right\}}{\rho_0 C_L(\alpha,\ \text{Mach})_I S_I} \left[\frac{\partial m_I(\Delta t_I)}{\partial \Delta t_I} + \frac{m_I(\Delta t_I)}{h_0}\left(\frac{\partial X_I(\Delta t_I)\cdot e_y}{\partial \Delta t_I}\right)\right], \qquad \text{(D19)}$$

$$m_I(\Delta t_I) = X(\Delta t_I) m_{I1}(\Delta t_I) + Y(\Delta t_I) m_{I2}(\Delta t_I) + Z(\Delta t_I) m_{I3}(\Delta t_I) \quad . \qquad \text{(D20)}$$

For the type I interceptor, the equations for $m_{I1}(\Delta t_I)$ are provided in Appendix B.

$$m_{I11}(t) = 230.5(1 - 0.449t) + (m_{I1}^0 - 33.59) \quad , \quad 0 \le t \le 1.61 \quad , \qquad \text{(D21)}$$

$$m_{I12}(t) = 85.91(1 - 0.159t) + (m_{I1}^0 - 33.59) \quad , \quad 1.61 \le t \le 3.83 \quad , \qquad \text{(D22)}$$

$$m_{I13}(t) = m_{I1}^0 \quad , \quad 3.83 \le t \ (t \text{ in seconds, } m_{I1}^0 \text{ typically 33.59 slugs}) \quad , \qquad \text{(D23)}$$

$$\frac{\partial m_{I1}(\Delta t_{I1})}{\partial \Delta t_{I1}} = \begin{cases} -103\ , & 0 \le \Delta t_{I1} \le 1.61 \\ -13.7\ , & 1.61 \le \Delta t_{I1} \le 3.83 \\ 0\ , & 3.83 \le \Delta t_{I1} \end{cases} \quad . \qquad \text{(D24)}$$

For the type II interceptor, the equations for $m_{I2}(\Delta t_{I1})$ are in Appendix B.

$$m_{I21}(t) = 218.8(1 - 0.714t) + (m_{I2}^0 - 15.62) \quad , \quad 0 \le t \le 1 \quad , \qquad \text{(D25)}$$

$$m_{I22}(t) = 109.4(1 - 0.429t) + (m_{I2}^0 - 15.62) \quad , \quad 1 \le t \le 2 \quad , \qquad \text{(D26)}$$

$$m_{I23}(t) = m_{I2}^0 \quad , \quad 2 \le t \quad , \qquad \text{(D27)}$$

$$\frac{\partial m_{I2}(\Delta t_{I2})}{\partial \Delta t_{I2}} = \begin{cases} -156\ , & 0 \le \Delta t_{I2} \le 1.0 \\ -46.9\ , & 1.0 \le \Delta t_{I2} \le 2.0 \\ 0 & 2.0 \le \Delta t_{I2} \end{cases} \quad . \qquad \text{(D28)}$$

(Note: $\Delta t_{M2} = t_{\text{wait}} + \Delta t_{I2}$.)

$$\frac{\partial X_I(\Delta t_I)}{\partial(\Delta t_I)} = |V_I(\Delta t_I)| e_{V_I^0} \quad , \quad e_{V_I^0} \triangleq e_{\Delta X_I} \quad . \qquad \text{(D29)}$$

D.6.

$$\Delta R(t_{\text{wait}}, \Delta t_I) = R_{MI}^0 - \int_0^{\Delta t_M} |V_M(\tau)| d\tau - \int_0^{\Delta t_I} V_I(\tau) d\tau \quad , \qquad \text{(D30)}$$

$$R_{MI}^0 \triangleq d_{MC}^0 + d_{CI}^0 \quad , \qquad \text{(D31)}$$

$$d_{MC}^0 \triangleq |X_M^0 - X_C| \quad , \quad d_{CI}^0 \triangleq |X_C - X_I^0| \quad , \qquad \text{(D32)}$$

$$\frac{\partial \Delta R}{\partial \Delta t_I} = -[|V_M(\Delta t_M)| + |V_I(\Delta t_I)|] \quad . \qquad \text{(D33)}$$

APPENDIX E

Derivatives for Using DBRENT [10] to Minimize the Function $|\Delta \mathbf{r}_{\mathbf{I}}(\phi_{\mathbf{M}})|$

Recall from the main text (Chapter 3), that we need to find the value of ϕ_M at which $\Delta r_I(\phi_M) = |r'_I(\phi_M)| - |r_I(\phi_M)|$ is a minimum. Having the derivatives of $\Delta r_I(\phi_M)$ considerably speeds up the process of finding this minimum (see Appendix G).

E.1.

$$\Delta r'_I(\phi_M) = \left(|r'_I\left(\phi_M\right)|\right)' - \left(|r_I\left(\phi_M\right)|\right)' \quad . \tag{E1}$$

E.2.

$$r'_I(\phi_M) = A(\phi_M)e_{x_M} + B(\phi_M)e_{y_M} \quad ,$$

$$|r'_I(\phi_M)| = \left[\left(A\left(\phi_M\right)\right)^2 + \left(B\left(\phi_M\right)\right)^2\right]^{1/2} \quad , \tag{E2}$$

$$(r'_I)' = \frac{dr'_I}{d\phi_M} = \frac{\partial r'_I}{\partial A}\frac{\partial A}{\partial \phi_M} + \frac{\partial r'_I}{\partial B}\frac{\partial B}{\partial \phi_M} \quad . \tag{E3}$$

E.3.

$$\frac{\partial r'_I}{\partial A} = [A(\phi_M)^2 + B(\phi_M)^2]^{-1/2}A(\phi_M) \quad , \tag{E4}$$

$$\frac{\partial r'_I}{\partial B} = [A(\phi_M)^2 + B(\phi_M)^2]^{-1/2}B(\phi_M) \quad , \tag{E5}$$

where

$$\begin{aligned} A(\phi_M) = {} & |r_M(\phi_M)| \sin(\phi_M) - d_{MC}(t_{\text{wait}}, \Delta t_I) \\ & - d_{CI}(t_{\text{wait}}, \Delta t_I) \cos(\phi_c^0) - r_I(\Delta t_I)\,(\sin\theta_c^0) \quad , \end{aligned} \tag{E6}$$

$$\begin{aligned} B(\phi_M) = {} & |r_M(\phi_M)|\,(\cos\phi_M) - r_M(\Delta t_M) - d_{CI}(t_{\text{wait}}, \Delta t_I)\,(\sin\theta_c^0) \\ & + r_I(\Delta t_I)\,(\cos\theta_c^0) \quad , \end{aligned} \tag{E7}$$

$$|r_M(\phi_M)| = \min\left\{ \frac{2m_M \exp\left\{\dfrac{X_M(\Delta t_M)\cdot e_y}{h_0} + \dfrac{\Delta h_M(\Delta t_M,\, \phi_M)}{h_0} - 1\right\}}{\rho_0 C_L(\alpha,\, \text{Mach})_M S_M} \quad , \quad 10^6 \right\} \tag{E8}$$

$(\Delta t_M$ fixed$)$.

$$\Delta h_M(\Delta t_M, \phi_M) = e_y \cdot \Big[(R_{z_M y_M x_M}\,(\alpha, \beta, \theta)\,|r_M\,(\Delta t_M)\,|) \times ((\sin\phi_M)\,e_{x_M} + (1-\cos\phi_M)\,e_{y_M})\Big] \quad , \qquad (\text{E9})$$

$$R_{z_M y_M x_M}\,(\gamma, \beta, \theta) \overset{\triangle}{=} R_{z_M}(\gamma) \cdot R_{y_M}(\beta) \cdot R_{x_M}(\theta) \quad , \qquad (\text{E10})$$

$$r_M(\Delta t_M) \equiv r_M(\Delta t_M, \phi_M)\Big|_{\phi_M = 0} \quad . \qquad (\text{E11})$$

$$|r_I(\Delta t_I)| = \min \left\{ \frac{2 m_I(\Delta t_I)\, \exp\left\{ \dfrac{X_I(\Delta t_I) \cdot e_y}{h_0} - 1 \right\}}{\rho_0 C_L(\alpha, \text{Mach})_I S_I} \quad , \quad 10^6 \right\} \quad . \qquad (\text{E12})$$

E.4.

$$\frac{\partial A(\phi_M)}{\partial \phi_M} = (\sin\phi_M) \frac{\partial |r_M(\phi_M)|}{\partial \phi_M} + |r_M(\phi_M)|\,(\cos\phi_M) \quad , \qquad (\text{E13})$$

$$\frac{\partial r_M(\phi_M)}{\partial \phi_M} = \frac{|r_M(\phi_M)|}{h_0} \, \frac{\partial \Delta h_M(\phi_M)}{\partial \phi_M} \quad , \qquad (\text{E14})$$

$$\frac{\partial \Delta h_M}{\partial \phi_M} = e_y \cdot [R_{z_M y_M x_M}(\gamma, \beta, \theta) |r_m(\Delta t_M)|\; ((\cos\phi_M)\,e_{x_M} + (\sin\phi_M)\,e_{y_M})] \,. (\text{E15})$$

E.5.

$$\frac{\partial B(\phi_M)}{\partial \phi_M} = (\cos\phi_M) \frac{\partial |r_M(\phi_M)|}{\partial \phi_M} - |r_M(\phi_M)|\,(\sin\phi_M) \quad . \qquad (\text{E16})$$

Note: $r_M(\phi_M) \equiv r_M(\Delta t_M, \phi_M.)$

E.6.

$$r_I(\Delta t_I) = r_I(\Delta t_I, \phi_I)\Big|_{\phi_I = 0} \quad , \qquad (\text{E17})$$

$$|r_I(\phi_M)| = \min \left\{ \frac{2 m_I(\Delta t_I)\, \exp\left\{ \dfrac{X_I(\Delta t_I) \cdot e_y}{h_0} + \dfrac{\Delta h_I\,(\Delta t_I, \phi_I\,(\phi_M))}{h_0} - 1 \right\}}{\rho_0 C_L(\alpha, \text{Mach})_I S_I} , 10^6 \right\} . \qquad (\text{E18})$$

$$\Delta h_I\,(\Delta t_I,\,\phi_I\,(\phi_M)) \\ = \Big[|r_I|\,(1-\cos\phi_I\,(\phi_M))\,e_{z_M}\times e_{V_I^0}+|r_I|\,(\sin\phi_I\,(\phi_M))\,e_{V_I^0}\Big]\bullet e_y\ , \qquad \text{(E19)}$$

$$\phi_I(\phi_M)=\cos^{-1}\left[\frac{B(\phi_M)\,\cos\theta_c^0-A(\phi_M)\,\sin\theta_c^0}{|r_I'(\phi_M)|}\right]\quad , \qquad \text{(E20)}$$

$$\frac{\partial|r_I(\phi_M)|}{\partial(\phi_M)}=\frac{|r_I(\phi_M)|}{h_0}\,\frac{\partial\Delta h_I\,(\phi_I\,(\phi_M))}{\partial\phi_I}\,\frac{\partial\phi_I(\phi_M)}{\partial\phi_M}\quad , \qquad \text{(E21)}$$

$$\frac{\partial h_I}{\partial\phi_I}=|r_I|\,\Big((\sin\phi_I)\,e_{z_M}\times e_{V_I^0}+(\cos\phi_I)\,e_{V_I^0}\Big)\bullet e_y\quad , \qquad \text{(E22)}$$

$$\frac{\partial\phi_I\phi_M)}{\partial\phi_M}=\frac{\partial}{\partial\phi_M}\,(\arccos\mathcal{U}\,(\phi_M))=-\frac{1}{\left(1-\mathcal{U}\,(\phi_M)^2\right)^{1/2}}\,\frac{\partial\mathcal{U}(\phi_M)}{\partial\phi_M}\quad ,$$

$$0\le\arccos\mathcal{U}\,(\phi_M)\le\pi\quad . \qquad \text{(E23)}$$

$$\mathcal{U}(\phi_M)=\frac{B(\phi_M)\,\cos\theta_c^0-A(\phi_M)\,\sin\theta_c^0}{|r_I'|(\phi_M)}=\frac{V(\phi_M)}{W(\phi_M)}\quad , \qquad \text{(E24)}$$

$$W(\phi_M)\overset{\triangle}{=}|r_I'|(\phi_M)=[A^2(\phi_M)+B^2(\phi_M)]^{1/2}\quad , \qquad \text{(E25)}$$

$$\frac{\partial\mathcal{U}(\phi_M)}{\partial\phi_M}=\frac{\partial\left(\frac{V}{W}\,(\phi_M)\right)}{\partial\phi_M}=\frac{1}{W(\phi_M)}\,\frac{\partial V(\phi_M)}{\partial\phi_M}-\frac{V(\phi_M)}{W(\phi_M)^2}\,\frac{\partial W(\phi_M)}{\partial\phi_M}\ , \qquad \text{(E26)}$$

$$V(\phi_M)=B(\phi_M)\,\cos\theta_c^0-A(\phi_M)\,\sin\theta_c^0\quad , \qquad \text{(E27)}$$

$$\frac{\partial V(\phi_M)}{\partial\phi_M}=\left(\cos\theta_c^0\right)\frac{\partial B(\phi_M)}{\partial\phi_M}-\left(\sin\theta_c^0\right)\frac{\partial A(\phi_M)}{\partial\phi_M}\quad , \qquad \text{(E28)}$$

$$\frac{\partial W(\phi_M)}{\partial\phi_M}=\frac{A(\phi_M)\dfrac{\partial A(\phi_M)}{\partial\phi_M}+B(\phi_M)\dfrac{\partial B(\phi_M)}{\partial\phi_M}}{\left(A^2\,(\phi_M)+B^2\,(\phi_M)\right)^{1/2}}\quad . \qquad \text{(E29)}$$

APPENDIX F

Some Derivatives Needed to Compute the Intercept Point Using RTSAFE [10]

Problem

"Solve

$$F(t_{\text{wait}}, t_c) = 0 \text{ for } t_c \quad , \qquad t_{\text{wait}} \text{ fixed."} \tag{F1}$$

$$F(t_{\text{wait}}, t_c) = \int_0^{t_c} |V_I(\tau)| d\tau - [U_x(t_{cw})^2 + U_y(t_{cw})^2 + U_z(t_{cw})^2]^{1/2} \quad , \tag{F2}$$

$$U_x(t_{cw}) = X^0_{MIx} + \int_0^{t_{cw}} \frac{V^0_{Mx}|V_M(\tau)|d\tau}{|V^0_M|} \quad , \tag{F3}$$

$$U_y(t_{cw}) = X^0_{MIy} + \int_0^{t_{cw}} \frac{V^0_{My}|V_M(\tau)|d\tau}{|V^0_M|} \quad , \tag{F4}$$

$$U_z(t_{cw}) = X^0_{MIz} + \int_0^{t_{cw}} \frac{V^0_{Mz}|V_M(\tau)|d\tau}{|V^0_M|} \quad , \tag{F5}$$

$$|V_M(\tau)| = \begin{cases} |V^0_M| \; , & 0 \le \tau \le \tau_0 \\ |V^0_M| \left[1 - \dfrac{\tau - \tau_0}{k_{DM}}\right] \; , & 0 \le \tau_0 \le \tau \\ |V^0_M| \left[1 - \dfrac{\tau}{k_{DM}}\right] \; , & \tau_0 < 0 \le \tau \end{cases} \quad . \tag{F6}$$

F.1.

$$\frac{\partial F(t_{\text{wait}}, t_c)}{\partial t_c} = \sum_{i=1}^{n} \frac{\partial F}{\partial \mathcal{U}_i} \frac{\partial \mathcal{U}_i(t_{\text{wait}}, t_c)}{\partial t_c} \quad , \quad n = 4 \quad . \tag{F7}$$

F.2.

$$\mathcal{U}_1(t_{\text{wait}}, t_c) = \int_0^{t_c} |V_I(\tau)| d\tau \; , \quad \frac{\partial F(t_{\text{wait}}, t_c)}{\partial \mathcal{U}_1} = 1 \tag{F8}$$

$$\frac{\partial \mathcal{U}_1(t_{\text{wait}}, t_c)}{\partial t_c} = |V_I(t_c)| \quad . \tag{F9}$$

F.3.

$$\mathcal{U}_2(t_c) = U_x(t_{cw}) \quad , \tag{F10}$$

$$\frac{\partial F}{\partial \mathcal{U}_2} = -A^{-1/2} U_x(t_{cw}) \quad , \quad A = U_x(t_{cw})^2 + U_y(t_{cw})^2 + U_z(t_{cw})^2 \quad , \tag{F11}$$

$$\frac{\partial \mathcal{U}_2}{\partial t_c} = \frac{V^0_{Mx}}{|V^0_M|} |V_M(t_{cw})| \quad . \tag{F12}$$

F.4.

$$\mathcal{U}_3(t_{cw}) = U_y(t_{cw}) \quad , \tag{F13}$$

$$\frac{\partial F}{\partial \mathcal{U}_3} = -A^{-1/2} U_y(t_{cw}) \quad , \tag{F14}$$

$$\frac{\partial \mathcal{U}_3}{\partial t_{cw}} = \frac{V^0_{My}}{|V^0_M|} |V_M(t_{cw})| \quad . \tag{F15}$$

F.5.

$$\mathcal{U}_4 = U_z(t_{cw}) \quad , \tag{F16}$$

$$\frac{\partial F}{\partial \mathcal{U}_4} = -A^{-1/2} U_z(t_{cw}) \quad , \quad \frac{\partial \mathcal{U}_4}{\partial t_{cw}} = \frac{V^0_{Mz}}{|V^0_M|} |V_M(t_{cw})| \quad . \tag{F17}$$

F.6.

$$\frac{\partial F(t_{\text{wait}}, t_c)}{\partial t_c}$$

$$= |V_I(t_c)| - \frac{A^{-1/2}}{|V^0_M|} \left[U_x(t_{cw}) V^0_{Mx} + U_y(t_{cw}) V^0_{My} + U_z(t_{cw}) V^0_{Mz} \right] |V_M(t_{cw})| \quad , \tag{F18}$$

$$A = [U_x(t_{cw})^2 + U_y(t_{cw})^2 + U_z(t_{cw})^2] \quad , \tag{F19}$$

$$|V_M(t_{cw})| = \begin{cases} |V^0_M| \ , & 0 \le t_{cw} \le \tau_0 \quad , \\ |V^0_M| \left(1 - \dfrac{t_{cw} - \tau_0}{k_{DM}}\right) \ , & 0 \le \tau_0 \le t_{cw} \quad , \\ |V^0_M| \left(1 - \dfrac{t_{cw}}{k_{DM}}\right) \ , & \tau_0 < 0 \le t_{cw} \quad , \end{cases} \tag{F20}$$

$$\tau_0 = \begin{cases} \dfrac{H_0 - X_M^0\bullet e_y}{V_M^0\bullet e_y} \,, & |V_M^0\bullet e_y| \geq 10^{-3} \quad , \\ 10^4 \,, & 0 \leq \ |V_M^0\bullet e_y| \leq 10^{-3} \,, \qquad H_0 \leq X_M^0\bullet e_y \quad , \\ -10^4 \,, & 0 \leq \ |V_M^0\bullet e_y| \leq 10^{-3} \,, \qquad H_0 > X_M^0\bullet e_y \quad , \end{cases} \tag{F21}$$

$$k_{DM} = \begin{cases} \dfrac{-2V_M^0 \, \min\{H_0, \, X_M^0\bullet e_y\}}{e_{V_M{}^0}\bullet e_y(V_M^0 - V_M)(V_M^0 + V_M^F)} \,, & V_M^F \leq V_M^0 - 1 \text{ and } e_{V_M^0}\bullet e_y < -10^{-3} \\ 10^3 \,, & \text{otherwise} \end{cases} \; . \tag{F22}$$

F.7.

F.7.1.

$$\int_0^{t_{cw}} |V_M(\tau)| d\tau = V_M^0 t_{cw} \overset{\triangle}{=} X_{M1}(t_{cw}) \ , \ 0 \leq t_{cw} \leq \tau_0 \quad . \tag{F23}$$

F.7.2.

$$\int_0^{t_{cw}} |V_M(\tau)| d\tau = V_M^0 \tau_0 + \int_{\tau_0}^{t_{cw}} V_M^0 \left(1 - (\tau - \tau_0)\, k\right) d\tau \tag{F24}$$

$$= V_M^0 \tau_0 + V_M^0 \tau \Big|_{\tau_0}^{t_{cw}} - \frac{V_M^0 k \tau^2}{2} \Big|_{\tau_0}^{t_{cw}} + V_M^0 k \tau_0 \tau \Big|_{\tau_0}^{t_{cw}} \tag{F25}$$

$$= V_M^0 t_{cw} + V_M^0 k \tau_0 t_{cw} - V_M^0 k \tau_0^2 - \frac{V_M^0 k t_{cw}^2}{2} + \frac{V_M^0 k \tau_0^2}{2} \tag{F26}$$

$$= V_M^0 \left[t_{cw} + k \tau_0 t_{cw} - \frac{k \tau_0^2}{2} - \frac{k t_{cw}^2}{2} \right] \overset{\triangle}{=} X_{M2}(t_{cw}) \quad , \qquad 0 \leq \tau_0 \leq t_{cw}. \tag{F27}$$

F.7.3.

$$\int_0^{t_{cw}} |V_M(\tau)| d\tau = V_M^0 \left[t_{cw} - \frac{k t_{cw}^2}{2} \right] \overset{\triangle}{=} X_{M3}(t_{cw}) \ , \ \tau_0 < 0 \leq t_{cw} \quad , \tag{F28}$$

where $k \overset{\triangle}{=} k_{DM}^{-1}$.

F.8. Concluding

$$\frac{\partial F(t_{\text{wait}}, t_c)}{\partial t_c} = |V_I(t_c)| - A(t_{cw})^{-1/2}|V_M(t_{cw})| \left[\frac{X^0_{MIx}V^0_{Mx} + X^0_{MIy}V^0_{My} + X^0_{MIz}V^0_{Mz}}{|V^0_M|} + \int_0^{t_{cw}} |V_M(\tau)| d\tau \right] \quad . \tag{F29}$$

APPENDIX G

Newton-Raphson Method Using Derivatives

Numerical Recipes

Algorithm MD uses several standard algorithms to find the root of a nonlinear function $f(x)$, *i.e.*, to solve $f(x) = 0$, and to find the minimum of such a function. In this Appendix, we include discussions and listings of the "recipes" we have implemented in MD. These were extracted with minor modifications from Reference [10].

G.1. Computing the Root of $f(x)$

Since the derivatives of $F^0(t_{\text{wait}}, \Delta t_I)$ were derived analytically (see Appendix D), we employ the Newton-Raphson method using first derivatives (Function RTSAFE) to compute its roots.

Newton-Raphson Method Using Derivatives

Perhaps the most celebrated of all one-dimensional root-finding routines is *Newton's method*, also called the *Newton-Raphson method.* What distinguishes this method from others is the the fact that it requires the evaluation of both the function $f(x)$, *and* the derivative $f'(x)$, at arbitrary points x. Geometrically, the Newton-Raphson formula consists of extending the tangent line at a current point x_i until it crosses zero, then setting the next guess x_{i+1} to the abscissa of the zero-crossing (see Figure G.1). Algebraically, the method derives from the familiar Taylor series expansion of a function in the neighborhood of a point.

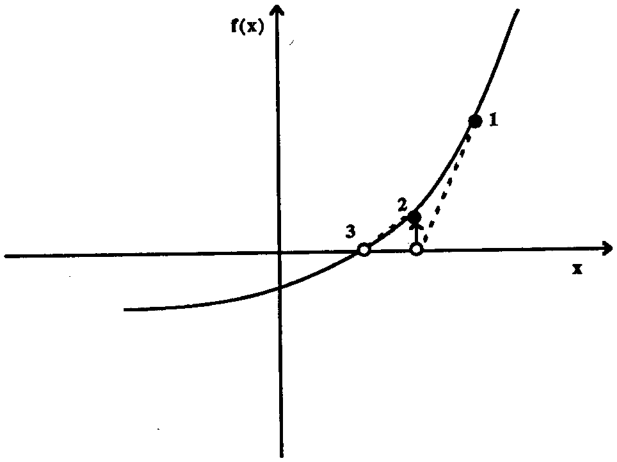


Figure G.1. Newton's method extrapolates the local derivative to find the next estimate of the root. In this example it works well and converges quadratically.

$$f(x+\delta) \approx f(x) + f'(x)\delta + \frac{f''(x)}{2}\delta^2 + \dots \quad . \tag{G1}$$

For small enough values of δ, and for well-behaved functions, the terms beyond linear are unimportant, hence $f(x+\delta) = 0$ implies

$$\delta = -\frac{f(x)}{f'(x)} \quad . \tag{G2}$$

Newton-Raphson is not restricted to one dimension. The method readily generalizes to multiple dimensions, as we shall see in the discussion to follow.

Far from a root, where the higher order terms in the series *are* important, the Newton-Raphson formula can give grossly inaccurate, meaningless corrections. For instance, the initial guess for the root might be so far from the true root as to let the search interval include a local maximum or minimum of the function. This can be death to the method (see Figure G.2). If an iteration places a trial guess near such a local extremum, so that the first derivative nearly vanishes, then Newton-Raphson sends its solution off to limbo, with vanishingly small hope of recovery. Like most powerful tools, Newton-Raphson can be destructive when used in inappropriate circumstances. Figure G.3 demonstrates another possible pathology.

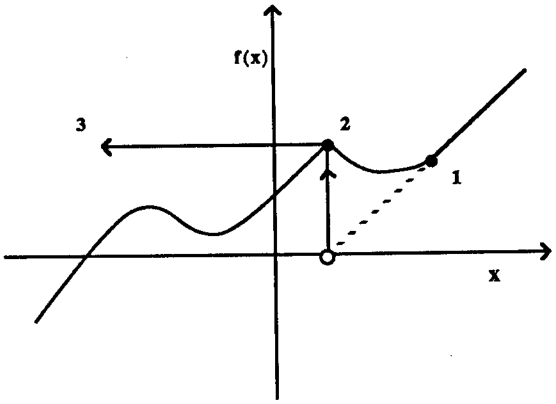


Figure G.2. Unfortunate case where Newton's method encounters a local extremum and shoots off to outer space. Here bracketing bounds, as in RTSAFE, would save the day.

Why do we call Newton-Raphson powerful? The answer lies in its rate of convergence. Within a small distance ϵ of x the function and its derivative are approximately:

$$\begin{aligned} f(x+\epsilon) &= f(x) + \epsilon f'(x) + \epsilon^2 \frac{f''(x)}{2} + \cdots \quad , \\ f'(x+\epsilon) &= f'(x) + \epsilon f''(x) + \cdots \quad . \end{aligned} \tag{G3}$$

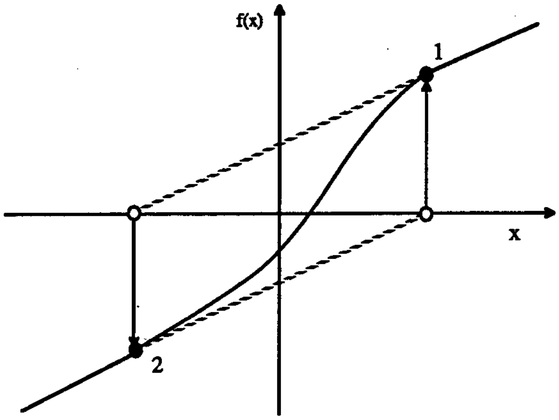


Figure G.3. Unfortunate case where Newton's method enters a nonconvergent cycle. This behavior is often encountered when the function f is obtained, in whole or in part, by table interpolation. With a better initial guess, the method would have succeeded.

By the Newton-Raphson formula.

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)} \quad , \tag{G4}$$

so that

$$\epsilon_{i+1} = \epsilon_i - \frac{f(x_i)}{f'(x_i)} \quad . \tag{G5}$$

When a trial solution x_i differs from the true root by ϵ_i, we can use (G3) to express $f(x_i)$, $f'(x_i)$ in (G4) in terms of ϵ_i and derivates at the root itself. The result is a recurrence relation for the deviations of the trial solutions,

$$\epsilon_{i+1} = -\epsilon_i^2 \frac{f''(x)}{2f'(x)} \quad . \tag{G6}$$

Equation (G6) says that Newton-Raphson converges *quadratically*. Near a root, the number of significant digits approximately *doubles* with each step. This very strong convergence property makes Newton-Raphson the method of choice for any function whose derivative can be evaluated efficiently and is continuous in the neighborhood of a root, and we have found it very effective in correcting locus errors Δ^* (see Section 3.4).

Even when Newton-Raphson is rejected for the early stages of convergence (because of its poor global convergence properties), it is very common to "polish up" a root with one or two steps of Newton-Raphson, which can multiply by two or four its number of significant figures!

For an efficient realization of Newton-Raphson the user provides a function subroutine which evaluates both $f(x)$ and its first derivative $f'(x)$ at the point x. The Newton-Raphson formula can also be applied using a numerical difference to approximate the true local derivative.

$$f'(x) \approx \frac{f(x+dx)-f(x)}{dx} \quad . \tag{G7}$$

This is not, however, a recommended procedure for the following reasons: (*i*) You are doing two function evaluations per step, so *at best* the superlinear order of convergence will be only $\sqrt{2}$. (*ii*) If you take dx too small you will be wiped out by roundoff, while if you take it too large your order of convergence will be only linear, no better than using the *initial* evaluation $f'(x_0)$ for all subsequent steps. Therefore, Newton-Raphson with numerical derivates is (in one dimension) always dominated by the secant method. (In multidimensions, where there is a paucity of available methods, Newton-Raphson with numerical derivates must be taken more seriously.)

The following subroutine calls a user-supplied subroutine FUNCD (X, FN, DF) which returns the function value as FN and the derivative as DF. We have included input bounds on the root simply to be consistent with previous root-finding routines: Newton does not adjust bounds, and works only on local information at the point X. The bounds are only used to pick the midpoint as the first guess, and to reject the solution if it wanders outside of the bounds.

FUNCTION RTNEWT(FUNCD,X1,X2,XACC)

> Using a the Newton-Raphson method, find the root of a function known to lie in the interval X1-X2. The root RTNEWT will be refined until its accuracy is known within ±XACC. FUNCD is a user-supplied subroutine that returns both the function value and the first derivative of the function at the point X.

```
PARAMETER   (JMAX=20)                          Set to maximum number of iterations.
RTNEWT=.5*(X1+X2)                              Initial guess.
DO 11  J=1,JMAX
         CALL  FUNCD(RTNEWT,F,DF)
         DX=F/DF
         RTNEWT=RTNEWT-DX
         IF((X1-RTNEWT)*(RTNEWT-X2).LT.0.)PAUSE  'jumped  out  of  brackets'
         IF(ABS(DX).LT.XACC)     RETURN Convergence.
  11  CONTINUE
PAUSE  'RTNEWT  exceeding  maximum  iterations'
END
```

While Newton-Raphson's global convergence properties are poor, it is fairly easy to design a fail-safe routine that utilizes a combination of bisection and Newton-Raphson. The hybrid algorithm takes a bisection step whenever Newton-Raphson would take the solution out of bounds, or whenever Newton-Raphson is not reducing the size of the brackets rapidly enough.

FUNCTION RTSAFE(FUNCD,X1,X2,XACC)

Using a combination of Newton-Raphson and bisection, find the root of a function bracketed between X1 and X2. The root, returned as the function value RTSAFE, will be refined until its accuracy is known within ±XACC. FUNCD is a user supplied subroutine which returns both the function value and the first derivative of the function.

```
PARAMETER     (MAXIT=100)                  Maximum allowed number of iterations.
CALL   FUNCD(X1,FL,DF)
CALL   FUNCD(X2,FH,DF)
IF(FL*FH.GE.0.)    PAUSE  'root  must  be  bracketed'
IF(FL.LT.0.)   THEN                        Orient the search so that f(XL) < 0.
      XL=X1
      XH=X2
ELSE
      XH=X1
      XL=X2
      SWAP=FL
      FL=FH
      FH=SWAP
ENDIF
RTSAFE=.5*(X1+X2)                          Initialize the guess for root,
DXOLD=ABS(X2-X1)                           the "step-size before last,"
DX=DXOLD                                   and the last step.
CALL  FUNCD(RTSAFE,F,DF)
DO 11 J=1,MAXIT                            Loop over allowed iterations
      IF(((RTSAFE-XH)*DF-F)*((RTSAFE-XL)*DF-F).GE.0. Bisect if Newton out of range,
 *          .OR.  ABS(2.*F).GT.ABS(DXOLD*DF)  )  THEN     or not decreasing fast enough.
           DXOLD=DX
           DX=0.5*(XH-XL)
           RTSAFE=XL+DX
           IF(XL.EQ.RTSAFE)   RETURN       Change in root is negligible.
      ELSE                                 Newton step acceptable.  Take it.
           DXOLD=DX
           DX=F/DF
           TEMP=RTSAFE
           RTSAFE=RTSAFE-DX
           IF(TEMP.EQ.RTSAFE)   RETURN
        ENDIF
      IF(ABS(DX).LT.XACC)   RETURN         Convergence criterion.
      CALL   FUNCD(RTSAFE,F,DF)            The one new function evaluation per iteration.
      IF(F.LT.0.)   THEN                   Maintain the bracket on the root.
           XL=RTSAFE
           FL=F
      ELSE
           XH=RTSAFE
           FH=F
      ENDIF
   11 CONTINUE
PAUSE  'RTSAFE  exceeding  maximum  iterations'
RETURN
END
```

For many functions the derivative $f'(x)$ often converges to machine accuracy before the function $f(x)$ itself does. When that is the case one need not subsequently update $f'(x)$. This shortcut is recommended only when you confidently understand the generic behavior of your function, but it speeds computations when the derivative calculation is laborious. (Formally this makes the convergence only linear, but if the derivative isn't changing anyway, you can do no better.) For further reading on this subject, see [G1–G3].

G.2. Minimizing a Function f(x)

Here we employ two recipes, DBRENT [G3], and BRENT [G3]. The first is used to compute the minimum $\min_{\phi_M}\{|\Delta r_I(\phi_M)|\}$, whose derivatives were analytically derived in Appendix E. The second is used to compute the value t^*_{wait} of t_{wait} for which the miss distance $MD(t_{\text{wait}})$ reaches a minimum. Here we do not have analytically determinable derivatives, and BRENT computes these as well as t^*_{wait}

G.3. Parabolic Interpolation and Brent's Method in One-Dimension

A golden section search is designed to handle, in effect, the worst possible case of function minimization, with the uncooperative minimum hunted down and cornered like a scared rabbit. But why assume the worst? If the function is nicely parabolic near to the minimum—surely the generic case for sufficiently smooth functions —then the parabola fitted through any three points ought to take us in a single leap to the minimum, or at least very near to it (see Figure G.4). Since we want to find an abscissa rather than an ordinate, the procedure is technically called *inverse parabolic interpolation.*

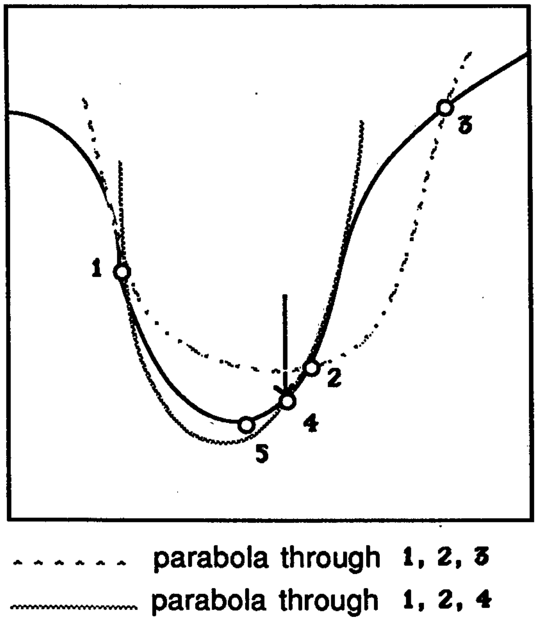


Figure G.4. Convergence to a minimum by inverse parabolic interpolation. A parabola (dashed line) is drawn through the three original points 1, 2, 3 on the given function (solid line). The function is evaluated at the parabola's minimum, 4, which replaces point 3. A new parabola (dotted line) is drawn through points 1, 4, 2. The minimum of this parabola is at 5, which is close to the minimum of the function.

The formula for the abscissa x which is the minimum of a parabola through three points $f(a)$, $f(b)$, and $f(c)$ is

$$x = b + \frac{1}{2}\,\frac{(b-a)^2[f(b)-f(c)] - (b-c)^2[f(b)-f(a)]}{(b-a)[f(b)-f(c)] - (b-c)[f(b)-f(a)]} \quad , \qquad \text{(G8)}$$

as you can easily derive. This formula fails only if the three points are collinear, in which case the denominator is zero (minimum of the parabola is infinitely far away). Note, however, that (G8) is as happy jumping to a parabolic maximum as to a minimum. No minimization scheme that depends solely on (G8) is likely to succeed in practice.

The exacting task is to invent a scheme which relies on a sure-but-slow technique, like golden section search, when the function is not cooperative, but which switches over to (G8) when the function allows. The task is nontrivial for several reasons, including these: (*i*) The housekeeping needed to avoid unnecessary function evaluations in switching between the two methods can be complicated. (*ii*) Careful attention must be given to the "endgame," where the function is being evaluated very near to the roundoff limit of equation (G8). (*iii*) The scheme for detecting a cooperative versus noncooperative function must be robust.

Brent's method [G4] is up to the task in all particulars. At any particular stage, it is keeping track of six function points (not necessarily all distinct), a, b, u, v, w, and x, defined as follows: the minimum is bracketed between a and b; x is the point with the very least function value found so far (or the most recent one in case of a tie); w is the point with the second least function value; v is the previous value of w; u is the point at which the function was evaluated most recently. Also appearing in the algorithm is the point x_m, the midpoint between a and b; however, the function is not evaluated there.

You can read the code below to understand the method's logical organization. Mention of a few general principles here may, however, be helpful: Parabolic interpolation is attempted, fitting through the points x, v, and w. To be acceptable, the parabolic step must (*i*) fall within the bounding interval (a, b), and (*ii*) imply a movement from the best current value x that is *less* than half the movement of the *step before last.* This second criterion insures that the parabolic steps are actually converging to something, rather than, say, bouncing around in some nonconvergent limit cycle. In the worst possible case, where the parabolic steps are acceptable but useless, the method will approximately alternate between parabolic steps and golden sections, converging in due course by virtue of the latter. The reason for comparing to the step *before* last seems essentially heuristic: experience shows that it is better not to "punish" the algorithm for a single bad step if it can make it up on the next one.

Another principle exemplified in the code is never to evaluate the function less than a distance `TOL` from a point already evaluated (or from a known bracketing point). The reason is that, as we saw in equation (G8), there is simply no information content in doing so: the function will differ from the value already evaluated only by an amount of order the roundoff error. Therefore in the code below you will find several tests and modifications of a potential new point, imposing this restriction. This restriction also interacts subtly with the test for "doneness," which the method takes into account.

A typical ending configuration for Brent's method is that a and b are $2 \times x \times$ TOL apart, with x (the best abscissa) at the midpoint of a and b, and therefore fractionally accurate to ±TOL.

Indulge us a final reminder that TOL should generally be no smaller than the square root of your machine's floating point precision.

FUNCTION BRENT(AX,BX,CX,F,TOL,XMIN)

> Given a function F, and given a bracketing triplet of abscissas **AX, BX, CX** (such that **BX** is between **AX** and **CX**, and **F(AX)** and **F(CX)**), this routine isolates the minimum to a fractional precision of about **TOL** using Brent's method. The abscissa of the minimum is returned as **XMIN**, and the minimum function value is returned as **BRENT**, the returned function value.

For further reading on this subject, see [G5].

G.4. One-Dimensional Search with First Derivatives

Here we want to accomplish precisely the same goal as in the previous section, namely to isolate a functional minimum that is bracketed by the triplet of abscissas $a < b < c$, but utilizing an additional capability to compute the function's first derivative as well as its value.

In principle, we might simply search for a zero of the derivative, ignoring the function value information, using a root finder like RTFLSP or ZBRENT [G3]. It doesn't take long to reject *that* idea: How do we distinguish maxima from minima? Where do we go from initial conditions where the derivatives on one or both of the outer bracketing points indicate that "downhill" is in the direction *out* of the bracketed interval?

We don't want to give up our strategy of maintaining a rigorous bracket on the minimum at all times. The only way to keep such a bracket is to update it using function (not derivative) information, with the central point in the bracketing triplet always that with the lowest function value. Therefore the role of the derivatives can only be to help

```
PARAMETER  (ITMAX=100,      CGOLD=.3819660,      ZEPS=1.0E-10)
```

Maximum allowed number of iterations, golden ratio, and a small number which protects against trying to achieve fractional accuracy for a minimum that happens to be exactly zero.

```
      A=MIN(AX,CX)                                       A and B must be in ascending order,
      B=MAX(AX,CX)                                       though the input abscissas need not be.
      V=BX                                               Initializations
      W=V
      X=V
      E=0                                                This will be the distance moved on the step before last.
      FX=F(X)
      FV=FX
      FW=FX
      DO 11  ITER=1,ITMAX                                Main program loop.
           XM=0.5*(A+B)
           TOL1=TOL*ABS(X)+ZEPS
           TOL2=2.*TOL1
           IF(ABS(X-XM).LE.(TOL2-.5*(B-A)))   GOTO   3   Test for done here.
           IF(ABS(E).GT.TOL1)   THEN                     Construct a trial parabolic fit.
                R=(X-W)*(FX-FV)
                Q=(X-V)*(FX-FW)
                P=(X-V)*Q-(X-W)*R
                Q=2.*(Q-R)
                IF(Q.GT.0.)   P=-P
                Q=ABS(Q)
                ETEMP=E
                E=D
                IF(ABS(P).GE.ABS(.5*Q*ETEMP)   .OR.   P.LE.Q*(A-X)   .OR.
 *                   P.GE.Q*(B-X))   GOTO  1
           The above conditions determine the acceptability of the parabolic fit.  Here it is o.k.
                D=P/Q                                    Take the parabolic step.
                U=X+D
                IF(U-A.LT.TOL2   .OR.   B-U.LT.TOL2)   D=SIGN(TOL1,XM-X)
                GOTO  2                                  Skip over the golden section step.
           ENDIF
 1         IF(X.GE.XM)   THEN                            We arrive here for a golden section step,
                E=A-X                                    which we take into the larger of the two segments.
           ELSE
                E=B-X
           ENDIF
           D=CGOLD*E                                     Take the golden section step.
 2         IF(ABS(D).GE.TOL1)   THEN                     Arrive here with D computed either from parabolic fit,
                U=X+D                                    or else from golden section.
            ELSE
                U=X+SIGN(TOL1,D)
           ENDIF
           FU=F(U)                                       This is the one function evaluation per iteration, and now
           IF(FU.LE.FX)   THEN                           we have to decide what to do with our function evaluation.
                  IF(U.GE.X)   THEN                      Housekeeping follows:
                     A=X
                ELSE
                     B=X
                ENDIF
                V=W
                FV=FW
                W=X
                FW=FX
                X=U
                FX=FU
           ELSE
                IF(U.LT.X)   THEN
                     A=U
                ELSE
                     B=U
                ENDIF
                IF(FU.LE.FW   .OR.   W.EQ.X)   THEN
                     V=W
                     FV=FW
                     W=U
                     FW=FU
                ELSE  IF(FU.LE.FV   .OR.   V.EQ.X   .OR.   V.EQ.W)   THEN
                     V=U
                     FV=FU
                ENDIF
           ENDIF                                         Done with housekeeping.  Back for another iteration.
    11  CONTINUE
      PAUSE  'Brent  exceed  maximum  iterations.'
 3    XMIN=X                                             Arrive here ready to exit with best values.
      BRENT=FX
      RETURN
      END
```

us choose new trial points within the bracket.

One school of thought is to "use everything you've got": Compute a polynomial of relatively high order (cubic or above) which agrees with some number of previous function and derivative evaluations. For example, there is a unique cubic that agrees with function and derivative at two points, and one can jump to the interpolated minimum of that cubic (if there is a minimum within the bracket). Suggested by Davidon and others, formulas for this tactic are given in Acton [G1].

We like to be more conservative than this. Once superlinear convergence sets in, it hardly matters whether its order is moderately lower or higher. In practical problems that we have met, most function evaluations are spent in getting globally close enough to the minimum for superlinear convergence to commence. So we are more worried about all the funny "stiff" things that high order polynomials can do and about their sensitivities to roundoff error.

This leads us to use derivative information only as follows: The sign of the derivative at the central point of the bracketing triplet $a < b < c$ indicates uniquely whether the next test point should be taken in the interval (a, b) or in the interval (b, c). The value of this derivative and of the derivative at the second-best-so-far point are extrapolated to zero by the secant method (inverse linear interpolation), which by itself is superlinear of order 1.618. (The golden mean again: See Acton, p. 57 [G1].) We impose the same sort of restrictions on this new trial point as in Brent's method. If the trial point must be rejected, we *bisect* the interval under scrutiny.

Observe that we have emphasized the use of derivative information in one-dimensional minimization. But beware of functions whose computed "derivatives" *don't* integrate up to the function value and *don't* accurately point the way to the minimum, usually because of roundoff errors, sometimes because of truncation error in the method of derivative evaluation.

You will see that the following routine is closely modeled on `BRENT` in the previous section.

FUNCTION DBRENT(AX, BX, CX, F, DF, TOL, XMIN)

Given a function F and its derivative function DF, and given a bracketing triplet of abscissas AX, BX, CX (such that BX is between AX and CX, and F(BX) is less than both F(AX) and F(CX)), this routine isolates the minimum to a fractional precision of about TOL using a modification of Brent's method that uses derivatives. The abscissa of the minimum is returned as XMIN, and the minimum function value is returned as DBRENT, the returned function value.

```
PARAMETER  (ITMAX=100,      ZEPS=1.0E-10)
    Comments following will point out only differences from the routine BRENT.  Read that routine first.

    LOGICAL  OK1,  OK2                                      Will be used as flags for whether proposed
    A=MIN(AX,CX)                                            steps are acceptable or not.
    B=MAX(AX,CX)
    V=BX
    W=V
    X=V
    E=0
    FX=F(X)
    FV=FX
    FW=FX
    DX=DF(X)                                                All our housekeeping chores are doubled by the necessity of
    DV=DX                                                   moving derivative values around as well as function values.
    DW=DX
    DO 11  ITER=1,ITMAX
         XM=0.5*(A+B)
    TOL1=TOL*ABS(X)+ZEPS
    TOL2=2.*TOL1
    IF(ABS(X-XM).LE.(TOL2-.5*(B-A)))  GOTO  3
    IF(ABS(E).GT.TOL1)  THEN
         D1=2.*(B-A)                                        Initialize these D's to an out-of-bracket value
         D2=D1
         IF(DW.NE.DX)   D1=(W-X)*DX/(DX-DW)                 Secant method.
         IF(DV.NE.DX)   D2=(V-X)*DX/(DX-DV)                 Secant method with the other stored point.
    Which of these two estimates of D shall we take?  We will insist that they be within the bracket,
    and on the side pointed to by the derivative at X.
         U1=X+D1
         U2=X+D2
         OK1=((A-U1)*(U1-B).GT.0.)   .AND.   (DX*D1.LE.0.)
         OK2=((A-U2)*(U2-B).GT.0.)   .AND.   (DX*D2.LE.0.)
         OLDE=E                                             Movement on the step before last.
         E=D                                                Take only an acceptable D, and if both are
         IF(.NOT.(OK1.OR.OK2))  THEN                        acceptable, then take the smallest one.
              GOTO 1
         ELSE IF  (OK1.AND.OK2)  THEN
              IF(ABS(D1).LT.ABS(D2))  THEN
                   D=D1
              ELSE
                   D=D2
              ENDIF
         ELSE IF  (OK1)  THEN
              D=D1
         ELSE
              D=D2
         ENDIF
         IF(ABS(D).GT.ABS(0.5*OLDE))  GOTO  1
         U=X+D
         IF(U-A.LT.TOL2  .OR.  B-U.LT.TOL2)  D=SIGN(TOL1,XM-X)
         GOTO 2
    ENDIF
1   IF(DX.GE.0.)  THEN                                      Decide which segment by the sign of the derivative.
         E=A-X
    ELSE
         E=B-X
    ENDIF
    D=0.5*E                                                 Bisect, not golden section.
2   IF(ABS(D).GE.TOL1)  THEN
         U=X+D
         FU=F(U)
```

```
      ELSE
          U=X+SIGN(TOL1,D)
          FU=F(U)
          IF(FU.GT.FX)  GOTO  3              If the minimum step in the downhill direction
      ENDIF                                  takes us uphill, then we are done.
      DU=DF(U)                               Now all the housekeeping, sigh.
      IF(FU.LE.FX)  THEN
          IF(U.GE.X)  THEN
              A=X
          ELSE
              B=X
          ENDIF
          V=W
          FV=FW
          DV=DW
          W=X
          FW=FX
          DW=DX
          X=U
          FX=FU
          DX=DU
          ELSE

              IF(U.LT.X)  THEN
                  A=U
              ELSE
                  B=U
              ENDIF
              IF(FU.LE.FW  .OR.  W.EQ.X)  THEN
                  V=W
                  FV=FW
                  DV=DW
                  W=U
                  FW=FU
                  DW=DU
              ELSE  IF(FU.LE.FV  .OR.  V.EQ.X  .OR.  V.EQ.W)  THEN
                  V=U
                  FV=FU
                  DV=DU
              ENDIF
          ENDIF
        11  CONTINUE
      PAUSE  'DBRENT  exceeded  maximum  iterations.'
   3  XMIN=X
      DBRENT=FX
      RETURN
      END
```

REFERENCES AND FURTHER READING:

[G1] Acton, Forman S. 1970, *Numerical Methods That Work* (New York: Harper and Row), Chapter 2.

[G2] Ralston, Anthony, and Rabinowitz, Philip. 1978, *A First Course in Numerical Analysis*, 2nd ed. (New York: McGraw-Hill).

[G3] Ortega, J., and Rheinboldt, W. 1970, *Iterative Solution of Nonlinear Equations in Several Variables* (New York: Academic Press).

[G4] Brent, Richard P. 1973, *Algorithms for Minimization without Derivatives* (Englewood Cliffs, N. J.: Prentice-Hall), Chapter 5.

[G5] Forsythe, George E., Malcolm, Michael A., and Moler, Cleve B. 1977, *Computer Methods for Mathematical Computations* (Englewood Cliffs, N.J.: Prentice-Hall), §8.2.

APPENDIX H

Solving for $t_w^{\pi/2}$

We solve

$$|d^0_{MC} e_{V^0_M}| = \int_0^{t^{\pi/2}_{IC}+t^{\pi/2}_w} |V_M(\tau)| d\tau \qquad \text{for } t^{\pi/2}_w \quad .$$

Define $d^0_{MC} = |X^0_M - X^{\pi/2}_c|$. There are three cases ($t^{\pi/2}_{IC}$ was found in Chapter 3).

H.1. Case 1. $X^0_M \bullet e_y > H_0$, $X^{\pi/2}_c \bullet e_y > H_0$.

This case is simple, since the MaRV speed $|V_M(\tau)| = |V^0_M|$, a constant.

Hence

$$t^{\pi/2}_w = \frac{(X_c - X^0_M)\bullet V^0_M}{(|V^0_M|)^2} - t^{\pi/2}_{IC} \quad , \tag{H1}$$

where $t^{\pi/2}_{IC}$ is the solution to the equation $|X^{\pi/2}_c - X^0_I| = \int_0^{t^{\pi/2}_{IC}} |V_I(\tau)| d\tau$, and $X^{\pi/2}_c$ was found in Chapter 3. Note that $t^{\pi/2}_w$ may be negative.

H.2. Case 2. $X^0_M \bullet e_y > H_0$, $X^{\pi/2}_c \bullet e_y \leq H_0$.

The MaRV speed $|V_M(\tau)|$ remains constant at $|V^0_M|$ until $\tau = \tau_0 = H_0 - X^0_M \bullet e_y / V^0_M \bullet e_y$ (Equation (2.4)). In τ_0 seconds, M travels $|V^0_M|\tau_0$ distance. Hence, all we need is to solve the following equation for γ:

$$R = |X^0_M - X^{\pi/2}_c| - |V^0_M|\tau_0 = \int_0^{\gamma} |V_M(\tau)| d\tau \quad , \tag{H2}$$

where

$$|V_M(\tau)| = |V^0_M| \left[1 - \frac{\tau}{k_{DM}}\right]$$

and

$$\gamma = t^{\pi/2}_{IC} + t^{\pi/2}_w - \tau_0 \quad .$$

We obtain

$$R = |V^0_M|\gamma - \frac{\gamma^2 |V^0_M|}{2k_{DM}} \quad , \tag{H3}$$

whose solution is:

$$\gamma = \frac{|V^0_M| - \sqrt{|V^0_M|^2 - \frac{2|V^0_M|R}{k_{DM}}}}{|V^0_M|/k_{DM}} = k_{DM}\left(1 - \sqrt{1 - \frac{2R}{|V^0_M|k_{DM}}}\right) \quad , \tag{H4}$$

and

$$t_w^{\pi/2} = k_{DM}\left(1 - \sqrt{1 - \frac{2R}{|V_M^0|k_{DM}}}\right) + \tau_0 - t_{IC}^{\pi/2} \quad , \tag{H5}$$

or

$$t_w^{\pi/2} = k_{DM}\left[1 - \sqrt{1 - \frac{2(|X_M^0 - X_c^{\pi/2}| - |V_M^0|\tau_0)}{|V_M^0|k_{DM}}}\right] + \tau_0 - t_{IC}^{\pi/2} \quad . \tag{H6}$$

H.3. Case 3. $X_M^0 \cdot e_y \leq H_0$, $X_c^{\pi/2} \cdot e_y \leq H_0$.

This case is similar to H.2: solve the following equation for $t_w^{\pi/2}$:

$$|X_M^0 - X_c^{\pi/2}| = \int_0^{t_{IC}^{\pi/2} + t_w^{\pi/2}} |V_M(\tau)|\, d\tau \quad . \tag{H7}$$

The solution is

$$t_w^{\pi/2} = k_{DM}\left(1 - \sqrt{1 - \frac{2(|X_M^0 - X_c^{\pi/2}|)}{|V_M^0|k_{DM}}}\right) - t_{IC}^{\pi/2} \quad . \tag{H8}$$

APPENDIX I

Some Derivatives for the Predictor-Corrector Step Δ^*

Have

$$r'_I(t_w,\, \Delta t_I,\, \phi^*_M) = A(\cdot)e_{x_M} + B(\cdot)e_{y_M} \quad , \tag{I1}$$

$$|r'_I| = (A^2 + B^2)^{1/2} \quad . \tag{I2}$$

Have $r_I\left(t_w,\, \Delta t_I,\, \phi^*_I\left(\phi^*_M\right)\right)$ for any t_w, Δt_I, ϕ^*_I

$$\phi^*_I = \cos^{-1}\left[\frac{-A\,\sin\theta_c^0 + B\,\cos\theta_c^0}{|r'_I(t_w,\, \Delta t_I,\, \phi^*_M)|}\right] \quad , \tag{I3}$$

$$\Delta(t_w,\, \Delta t_I,\, \phi_M) \overset{\triangle}{=} |r'_I(\cdot)| - |r_I(\cdot)| \quad , \tag{I4}$$

$$\Delta^*(t_w,\, \Delta t_I,\, \phi^*_M) = \min_{\phi_M}\left\{\Delta(t_w,\, \Delta t_I,\, \phi_M)\right\} \quad . \tag{I5}$$

I.1.

$$\frac{\partial\Delta^*}{\partial\Delta t_I} = \frac{\partial|r'_I|}{\partial\Delta t_I} - \frac{\partial|r_I|}{\partial\Delta t_I} \quad . \tag{I6}$$

I.2.

$$\frac{\partial|r'_I|}{\partial\Delta t_I} = (A^2 + B^2)^{-1/2}\left(A\frac{\partial A}{\partial\Delta t_I} + B\frac{\partial B}{\partial\Delta t_I}\right) \quad , \tag{I7}$$

$$\begin{aligned} A = {} & |r_M(\phi^*_M)|\,(\sin\phi^*_M) - d_{MC}(t_w,\, \Delta t_I) - d_{CI}(t_w,\, \Delta t_I)\,(\cos\theta_c^0) \\ & - |r_I(\Delta t_I)|\,(\sin\theta_c^0) \quad , \end{aligned} \tag{I8}$$

$$B = |r_M(\phi^*_M)|\,(\cos\phi^*_M) - |r_M(\Delta t_M)| - d_{CI}(t_w,\, \Delta t_I)\,(\sin\theta_c^0) + |r_I(\Delta t_I)|\,(\cos\theta_c^0) \ , \tag{I9}$$

$$d_{MC}(t_w,\, \Delta t_I) = |X_M^0 - X_c(t_w)| - \int_0^{\Delta t_I + t_w} |V_M(\tau)|\, d\tau \quad , \tag{I10}$$

$$d_{CI}(t_w,\, \Delta t_I) = |X_I^0 - X_c(t_w)| - \int_0^{\Delta t_I} |V_I(\tau)|\, d\tau \quad , \tag{I11}$$

$$\frac{\partial A}{\partial \Delta t_I} = (\sin \phi_M^*)\frac{\partial |r_M(\phi_M^*)|}{\partial \Delta t_I} + |V_M(\Delta t_I + t_w)|$$

$$+ (\cos \theta_c^0)|V_I(\Delta t_I)| - (\sin \theta_c^0)\frac{\partial |r_I(\Delta t_I)|}{\partial \Delta t_I} \quad , \tag{I12}$$

$$\frac{\partial |r_M(\phi_M^*)|}{\partial \Delta t_I} = \frac{|r_M(t_w,\, \Delta t_I,\, \phi_M^*)|}{h_0}\left(\frac{\partial X_M(t_w,\, \Delta t_I) \cdot e_y}{\partial \Delta t_I} + \frac{\partial \Delta h_M}{\partial \Delta t_I}(t_w,\, \Delta t_I,\, \phi_M)\right) , \tag{I13}$$

$$X_M(t_w,\, \Delta t_I) = X_M^0 + \int_0^{t_w + \Delta t_I} |V_M(\tau)|\, d\tau\, e_{V_M^0} \quad , \tag{I14}$$

$$\frac{\partial X_M(t_w,\, \Delta t_I)}{\partial \Delta t_I} = |V_M(t_w,\, \Delta t_I)| e_{V_M^0} \quad , \tag{I15}$$

$$\frac{\partial \Delta h_M(t_w,\, \Delta t_I,\, \phi_M)}{\partial \Delta t_I} = e_y \cdot R_{z_M y_M x_M}(\gamma,\, \beta,\, \theta)\Big[(\sin \phi_M) e_{x_M}$$

$$+ (1 - \cos \phi_M)\, e_{y_M}\Big]\frac{\partial r_M(t_w,\, \Delta t_I)}{\partial \Delta t_I} \quad , \tag{I16}$$

$$\frac{\partial |r_I(\Delta t_I)|}{\partial (\Delta t_I)} = |r_I(\Delta t_I)|\frac{\partial X_I(\Delta t_I) \cdot e_y}{\partial \Delta t_I h_0} \quad , \tag{I17}$$

$$X_I(\Delta t_I) = X_I^0 + \int_0^{\Delta t_I} |V_I(\tau)|\, d\tau\, e_{\Delta X_I} \quad , \tag{I18}$$

$$\frac{\partial X_I(\Delta t_I)}{\partial \Delta t_I} = |V_I(\Delta t_I)| e_{\Delta X_I} \quad , \tag{I19}$$

$$\frac{\partial B}{\partial \Delta t_I} = (\cos \phi_M^*)\frac{\partial |r_M(\phi_M^*)|}{\partial \Delta t_I} - \frac{\partial r_M(t_w,\, \Delta t_I)}{\partial \Delta t_I} + (\sin \theta_c^0)|V_I(\Delta t_I)|$$

$$+ (\cos \theta_c^0)\frac{\partial |r_I(\Delta t_I)|}{\partial \Delta t_I} \quad , \tag{I20}$$

$$\frac{\partial r_M(t_w,\, \Delta t_I)}{\partial \Delta t_I} = \frac{r_M(t_w,\, \Delta t_I)}{h_0}\frac{\partial X_M(t_w,\, \Delta t_I) \cdot e_y}{\partial \Delta t_I} \quad . \tag{I21}$$

I.3.

$$\frac{\partial |r_I|}{\partial \Delta t_I} = \frac{r_I(t_w,\, \Delta t_I,\, \phi_I)}{h_0}\left[\frac{\partial X_I(\Delta t_I)}{\partial \Delta t_I} \cdot e_y + \frac{\partial \Delta h_I(\Delta t_I,\, \phi_I)}{\partial \Delta t_I}\right] \quad , \tag{I22}$$

$$\Delta h_I(\Delta t_I) = \left[|r_I(\Delta t_I)| \, (1 - \cos\phi_I) \, \frac{e_{z_M} \times e_{V_I^0}}{|e_{z_m} \times e_{V_I^0}|} + (\sin\theta_I) \, e_{V_I^0} \right] \bullet e_y \quad , \tag{I23}$$

$$\phi_I = \cos^{-1}\left[\frac{B \, \cos\theta_c^0 - A \, \sin\theta_c^0}{|r_I'|} \right] \quad , \tag{I24}$$

$$\frac{\partial \Delta h_I(\Delta t_I)}{\partial \Delta t_I} = \frac{\partial \Delta h_I(\Delta t_I)}{\partial |r_I(\Delta t_I)|} \, \frac{\partial |r_I(\Delta t_I)|}{\partial \Delta t_I} + \frac{\partial \Delta h_I(\Delta t_I)}{\partial \phi_I} \, \frac{\partial \phi_I}{\partial \Delta t_I} \quad , \tag{I25}$$

$$\frac{\partial \Delta h_I(\Delta t_I)}{\partial |r_I(\Delta t_I)|} = \left[(1 - \cos\phi_I) \, \frac{e_{z_M} \times e_{V_I^0}}{|e_{z_M} \times e_{V_I^0}|} + (\sin\phi_I) e_{V_I{}^0} \right] \bullet e_y \quad , \tag{I26}$$

$$\frac{\partial |r_I(\Delta t_I)|}{\partial \Delta t_I} \qquad \text{was developed earlier (Eq. (D.19))} \quad , \tag{I27}$$

$$\frac{\partial \Delta h_I(\Delta t_I)}{\partial \phi_I} = |r_I(\Delta t_I)| e_y \bullet \left[(\sin\phi_I) \frac{e_{z_M} \times e_{V_I^0}}{|e_{z_M} \times e_{V_I^0}|} + (\cos\phi_I) e_{V_I{}^0} \right] \quad , \tag{I28}$$

$$\frac{\partial \phi_I}{\partial \Delta t_I} = \frac{\partial \phi_I}{\partial |r_I'|} \, \frac{\partial |r_I'|}{\partial \Delta t_I} + \frac{\partial \phi_I}{\partial B} \, \frac{\partial B}{\partial \Delta t_I} + \frac{\partial \phi_I}{\partial A} \, \frac{\partial A}{\partial \Delta t_I} \quad , \tag{I29}$$

$$\frac{\partial \phi_I}{\partial |r_I'|} = \frac{\partial}{\partial (r_I')} \, \text{arccos} \left(\mathcal{U} \left(|r_I'| \right) \right) = - \frac{1}{\left(1 - \mathcal{U} \left(|r_I'| \right)^2 \right)^{1/2}} \, \frac{\partial \mathcal{U}(|r_I'|)}{\partial |r_I'|} \quad ,$$

$$0 \leq \text{arccos} \left(\mathcal{U}(|r_I'|) \right) \leq \pi \quad , \tag{I30}$$

$$\mathcal{U}(|r_I'|) = \frac{B \, \cos\theta_c^0 - A \, \sin\theta_c^0}{|r_I'|} \quad , \tag{I31}$$

$$\frac{\partial \mathcal{U}(|r_I'|)}{\partial |r_I'|} = \frac{A \, \sin\theta_c^0 - B \, \cos\theta_c^0}{(|r_I'|)^2} \quad , \tag{I32}$$

$$\frac{\partial |r_I'|}{\partial \Delta t_I} \qquad \text{was derived earlier (Eqs. (I12), (I17), (I20))} \quad , \tag{I33}$$

$$\frac{\partial \phi_I}{\partial B} = - \frac{\cos\theta_c^0}{\left(1 - \mathcal{U} \left(|r_I'| \right)^2 \right)^{1/2} |r_I'|} \quad , \tag{I34}$$

$$\frac{\partial \phi_I}{\partial A} = \frac{\sin(\theta_c^0)}{\left(1 - \mathcal{U} \left(|r_I'| \right)^2 \right)^{1/2} |r_I'|} \quad , \tag{I35}$$

$$\frac{\partial B}{\partial \Delta t_I} \quad \text{derived earlier (Eqs. (I20))} \quad , \tag{I36}$$

$$\frac{\partial A}{\partial \Delta t_I} \quad \text{derived earlier (Eqs. (I12))} \quad . \tag{I37}$$

Correction Step:

$$\text{Error: } \Delta^* = \frac{\partial \Delta^*}{\partial \Delta t_I} \Delta t_I \quad . \tag{I38}$$

Choose correction step:

$$\Delta t_I = \frac{-\Delta^*}{\frac{\partial \Delta^*}{\partial \Delta t_I}} \quad . \tag{I39}$$

List of Symbols

α: Angle of attack

a_{I1} (a_{I2}): Acceleration of interceptor of type 1 (2)

a_k: kth asset

C_D: Drag coefficient

C_I: Instantaneous center of maximal interceptor turn

C_L: Lift coefficient

C_M: Instantaneous center of maximal MaRV turn

C_q: Costs incurred by allocating defense resources to O_q

$C(I_\alpha)$: Lifecycle cost of one interceptor of type α

δ: Interceptor and defense system delay time

Δ: Geometric distance between MaRV and interceptor loci (trajectories)

$\Delta\Delta t_I$: Size of Δt_I correction step used to bring MaRV and interceptor loci together

Δh_I: Change in altitude of interceptor I during banking displacement angle ϕ_I

Δh_M: Change in altitude experienced by MaRV when it turns through an angular displacement ϕ_M

$\Delta R_{MI}(t_w, \Delta t_I^0)$: Total distance traveled by both vehicles for waiting time t_w and interceptor flight time $\Delta t_I^0 (\Delta t_I^0 + t_w = \Delta t_M^0)$

Δt_I: Flight time of interceptor I

Δt_I^0: Interceptor flight time at which d_{MI}^0 is measured (flight time to reach turning point)

Δt_M: Flight time of MaRV since last state update

$d_{CI}(\Delta t_M^0)$: Distance from interception point X_c to interceptor turning position (at turning time $\Delta t_M^0 = \Delta t_I^0 + t_w$)

d_{MC}^0: Distance from current MaRV position to interception ("collision") point X_c

$d_{MC}(\Delta t_M^0)$: Distance from interception point X_c to MaRV turning position (at turning time Δt_M^0)

d^0_{MI}: Geometric estimate of MaRV-interceptor distance at banking (turning) time $\Delta t^0_M = \Delta t^0_I + t_w$ (for initialization purposes)

$e_{V^0_q}$ ($e_{V^0_M}$): Unit vector along V_q (V^0_M)

e_y: Unit vector along y-axis (altitude)

F: Vehicle centripetal force

f_q: Footprint of O_q (a random function)

(γ, β, θ): Rotation angles of (x_M, y_M, z_M) coordinates

H_0: Effective height of atmosphere

H_{ko}: Keep-out altitude

I_k: Interceptor pool associated with asset a_k

I_{ku}: uth interceptor in interceptor pool I_k

$\mathcal{K}$: Collection of assets

K: Quantity of assets

k_{DI1} (k_{DI2}): Deceleration constant of interceptor of type 1 (2)

k_{DM}: MaRV deceleration constant

$KR_{\alpha q}$: Kill radius of an interceptor of type α against threat O_q

L: Lift force

L_q: Loss incurred if object O_q leaks through

L^B_q: Leakage loss incurred if O_q is a BRV

L^M_q: Leakage loss incurred if O_q is a MaRV

$\mathcal{M}$: Collection of MaRVs

M: Mach number

M_j: jth MaRV

$MD_{\alpha q}$: Miss distance of interceptor of type α against threat O_q

m: Vehicle mass

$m_{\alpha q}$: Quantity of defense resources of type α allocated to O_q

$m_{\alpha qs}$: Number of interceptors of type α launched from site s towards O_q

$m^*_{\alpha q}$: Optimal value of variable $m_{\alpha q}$

m_I: Mass of interceptor I

m^0_{I1} (m^0_{I2}): Payload (net mass) of type 1 (type 2) interceptor

m_M: Mass of MaRV

O_q: qth threat object

ϕ_I: Banking (turning) angle (angular displacement) of interceptor I

ϕ^*_I: Interceptor angular displacement from start of turn to point of locus contact (Q^*)

ϕ_M: Banking or turning angle (angular displacement) of MaRV

ϕ^*_M: MaRV angular displacement from start of turn to point of locus contact (Q^*)

p^*_{Kq}: Probability that O_q will be killed if an interceptor is launched at the optimal time t^*_{wait}

p^0_{Kq}: Probability that O_q will be killed if an interceptor is launched now

p_{Lq}: Probability that O_q leaks through defense

p^B_{Lq}: Leakage probability if O_q is a BRV

p^M_{Lq}: Leakage probability if O_q is a MaRV

p_{Lq_s}: Probability that O_q will leak through if a single interceptor is launched from site s

p^D_q: Probability that O_q is a decoy

p^M_q: Probability that O_q is a MaRV

p^B_q: Probability that O_q is a BRV

p^M_{qk} (p^B_{qk}): Probability that O_q is headed for a_k if O_q is a MaRV (BRV)

Q: Total quantity of threat objects

Q_I: Index of the last threat object on the priority list for which a sufficient quantity of interceptors is available

Q^*: Point of locus contact

ρ: Atmospheric density

R_0: Extremal region

R_1: Exoatmospheric region

R_2: Intermediate region

R_3: Tailchase region

R_{MI}^0: Estimate of total distance separating vehicles at most recent update time t_0

$R_{z_M^A y_M^A x_M^A}$: MaRV rotation matrix between action coordinates and MaRV coordinates

r_I: Vector collinear with r_I' from interceptor turn center to a point on the interceptor locus

r_I': Vector from interceptor turn center to point Q on MaRV locus

$r_M(\Delta t_M)$: Minimum curvature radius vector of MaRV at $t = \Delta t_M$

r_q: Threat ranking ratio for object O_q

$|r_I|$: Minimum radius of curvature of interceptor I

$|r_M(t)|$: Curvature radius of MaRV at time t

S: Effective vehicle area

$S_{\alpha u}$: Collection of sites each of which contains at least one interceptor of type α

sign: Signum function

τ: Clock increment (clock "step time")

τ_0: Time remaining before reaching H_0

τ_q: Time remaining for O_q to reach H_{ko}

θ_c^0: Incidence (collision) angle of interceptor relative to MaRV

T_0^F: Total flying time of threat object before ground impact

t^*: Time at which interceptor reaches X_I^*

t_0^F: Flying time from H_0 to ground impact

t_B: Time when both vehicles start banking

t_{BM}^* (t_{BI}^*): Turning time required by MaRV (interceptor) to reach Q^*

t^F: Flying time of MaRV from point below H_0 to ground impact

t_M^* (t_I^*): MaRV (interceptor) turning time to reach Q^*

t_w, t_{wait}:	Waiting time
t^*_{wait}:	Optimal launch delay (waiting) time
t^0_w:	Initial waiting time (usually $t^0_w = 0$)
U_α:	Inventory size of type α resources
V:	Value function of assets (usage clear from context)
V:	Vehicle velocity (usage clear from context)
V^0_I:	Launch velocity of interceptor
V_{I1} (V_{I2}):	Velocity of interceptor of type 1 (2)
$V^0_q(V^0_M)$:	Current velocity of O_q (MaRV)
$V^F_q(V^F_M)$:	Velocity of O_q (MaRV) at impact.
$\|V\|$, s:	vehicle speed
W_i:	Opportunity window i ($i = 1, 2, 3$)
$X^{\pi/2}_c$:	Point on the extended MaRV trajectory closest to the interceptor launch location
X^{Tc}_c:	Interception point in the tailchase region
X^*_I:	Position of interceptor when reaching Q^* ($X^*_I = Q^*$)
X^0_I:	Launch location of interceptor I
X_{I1} (X_{I2}):	Displacement of interceptor of type 1 (2)
$X_M(t)$:	Position of MaRV at time t
$X^*_M(t^*)$:	Position of MaRV when $X^*_I = Q^*$
X^0_q:	Current position of O_q
(x_M, y_M, z_M):	MaRV coordinates (M-system)
(x^A_M, y^A_M, z^A_M):	Action coordinate system (A-system)